

| | | | |
|----------------|-----|--------------|------------------|
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLL |
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLL |
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDD | DDD | CCC | LLL |
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLLLLLLLLLLLLLLL |
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLLLLLLLLLLLLLLL |
| DDDDDDDDDDDDDD | | CCCCCCCCCCCC | LLLLLLLLLLLLLLLL |

SSSS

| | | |
|------|-----|---|
| (3) | 130 | ALLOCATE DEVICE |
| (4) | 271 | ASSIGN LOGICAL NAME TO EQUIVALENCE STRING |
| (5) | 357 | DEFINE LOGICAL NAME EQUIVALENCE |
| (7) | 482 | DEALLOCATE DEVICE |
| (8) | 515 | DEASSIGN LOGICAL NAME EQUIVALENCE |
| (9) | 563 | TEST IF LOGICAL NAME IS SYSS\$OUTPUT |
| (10) | 591 | PROCESS COMMON COMMAND QUALIFIERS |
| (11) | 740 | GET TRANSLATION ATTRIBUTES |
| (12) | 772 | CREATE LOGICAL NAME TABLE |
| (13) | 945 | SHOW LOGICAL NAME EQUIVALENCES |

```
0000 1 .TITLE LOGICAL - LOGICAL NAME COMMANDS
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6 *****
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 9 * ALL RIGHTS RESERVED. *
0000 10
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 16 * TRANSFERRED. *
0000 17
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 20 * CORPORATION. *
0000 21
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 24
0000 25 *****
0000 26 *****
0000 27
0000 28 LOGICAL NAME DCLS COMMAND EXECUTION
0000 29
0000 30 ALLOCATE DEVICE
0000 31 ASSIGN LOGICAL NAME
0000 32 DEALLOCATE DEVICE
0000 33 DEASSIGN LOGICAL NAME
0000 34 DEFINE LOGICAL NAME
0000 35 CREATE LOGICAL NAME TABLE
0000 36 SHOW LOGICAL NAME TRANSLATION
0000 37
0000 38 Peter George 20-April-1983
0000 39
0000 40 MODIFIED BY:
0000 41
0000 42 V03-007 HWS0078 Harold Schultz 02-Jul-1984
0000 43 Fix negation of table qualifiers in ASSIGN, DEASSIGN,
0000 44 and DEFINE commands.
0000 45
0000 46 V03-006 HWS0041 Harold Schultz 12-Apr-1984
0000 47 Add ALLOCATE /GENERIC.
0000 48
0000 49 V03-005 PCG0003 Peter George 20-Mar-1984
0000 50 Add /JOB qualifier.
0000 51
0000 52 V03-004 HWS0005 Harold Schultz 07-Feb-1984
0000 53 Added /PROTECTION=(SY:RWED,OW:RWED,...) qualifier for
0000 54 when creating a logical name table.
0000 55 Add /LOG qualifier when creating a logical name table.
0000 56 Output informational messages after table creation.
0000 57
```


| | | | | | | |
|------|----|------|---------|---------|--|-------------|
| 0000 | 58 | : | V03-003 | TMK0001 | Todd M. Katz | 12-Oct-1983 |
| 0000 | 59 | : | | | Translate logical names using LNMSDCL_LOGICAL as the table | |
| 0000 | 60 | : | | | name instead of LNMSDEFAULT_SEARCH. | |
| 0000 | 61 | : | | | | |
| 0000 | 62 | : | V03-002 | PCG0002 | Peter George | 01-Jul-1983 |
| 0000 | 63 | : | | | Fix bug in ALLOCATE command parsing. | |
| 0000 | 64 | : | | | Replace old logical name commands. | |
| 0000 | 65 | : | | | Stop fooling around with the CRELOG bit. | |
| 0000 | 66 | : | | | | |
| 0000 | 67 | : | V03-001 | PCG0001 | Peter George | 15-Jun-1983 |
| 0000 | 68 | : | | | Return more helpful status when ALLOCATE fails. | |
| 0000 | 69 | : | | | Pass equivalence name to DCL\$OPEN_OUTPUT. | |
| 0000 | 70 | :--- | | | | |

```
0000 72 : MACRO LIBRARY CALLS
0000 73 :
0000 74 :
0000 75 :
0000 76 PRCDEF ;DEFINE PROCESS WORK AREA
0000 77 WRKDEF ;DEFINE COMMAND WORK AREA
0000 78 PTRDEF ;DEFINE RESULT PARSE DESCRIPTOR FORMAT
0000 79 $CLMSGDEF ;DEFINE ERROR/STATUS VALUES
0000 80 $LNHDEF ;DEFINE LOGICAL NAME OFFSETS
0000 81 $PSLDEF ;DEFINE PROCESSOR STATUS FIELDS
0000 82 $SSDEF ;DEFINE SYSTEM STATUS VALUES
0000 83
00000000 84 .PSECT DCL$ZCODE,BYTE,RD,NOWRT
0000 85
0000 86 : LOCAL DATA
0000 87 :
0000 88 :
0000 89 OUTPUTNAM:
54 55 50 54 55 4F 24 53 59 53 00' 0000 90 .ASCIC 'SYSSOUTPUT'
OA 0000
000B 91 LNMSPROCESS:
53 53 45 43 4F 52 50 24 4D 4E 4C 00' 000B 92 .ASCIC 'LNMSPROCESS'
OB 000B
0017 93 LNMSJOB:
42 4F 4A 24 4D 4E 4C 00' 0017 94 .ASCIC 'LNMSJOB'
07 0017
001F 95 LNMSGROUP:
50 55 4F 52 47 24 4D 4E 4C 00' 001F 96 .ASCIC 'LNMSGROUP'
09 001F
0029 97 LNMS$SYSTEM:
4D 45 54 53 59 53 24 4D 4E 4C 00' 0029 98 .ASCIC 'LNMS$SYSTEM'
OA 0029
0034 99 LNMSDCL_LOGICAL:
47 4F 4C 5F 4C 43 44 24 4D 4E 4C 00' 0034 100 .ASCIC 'LNMSDCL_LOGICAL'
4C 41 43 49 0040
OF 0034
0044 101 LNMSPROCESS_DIRECTORY:
53 53 45 43 4F 52 50 24 4D 4E 4C 00' 0044 102 .ASCIC 'LNMSPROCESS_DIRECTORY'
59 52 4F 54 43 45 52 49 44 5F 0050
15 0044
005A 103 LNMSFILE_DEV:
45 44 5F 45 4C 49 46 24 4D 4E 4C 00' 005A 104 .ASCIC 'LNMSFILE_DEV'
56 0066
OC 005A
0067 105 UNDEFINED:
44 45 4E 49 46 45 44 4E 55 00' 0067 106 .ASCIC 'UNDEFINED'
09 0067
0071 107 LOGICALMSG:
41 21 22 20 3D 20 53 41 21 20 20 00' 0071 108 .ASCIC ' !AS = "!AS" (!AS)'
29 53 41 21 28 20 20 22 53 007D
14 0071
52 57 45 44 0086 109 ACCESS: .ASCII /DEWR/ ;ACCESS PROTECTION CODES
53 4F 47 57 008A 110 CLASS: .ASCII /WGOS/ ;PROTECTION CLASSES
008E 111 :
008E 112 : DEFINE OFFSETS FOR COMMON PARSING DATA STRUCTURE
008E 113 :
00000020 008E 114 EQUAM = 32
```

LOGICAL
V04-000

- LOGICAL NAME COMMANDS

F 15

16-SEP-1984 00:08:00 VAX/VMS Macro V04-00
4-SEP-1984 23:41:57 [DCL.SRC]LOGICAL.MAR;1

Page 4
(2)

| | | | |
|----------|------|-----|-------------------|
| 00000018 | 008E | 115 | LOGNAM = 24 |
| 00000010 | 008E | 116 | TABNAM = 16 |
| 0000000C | 008E | 117 | ACMODE = 12 |
| 00000008 | 008E | 118 | QUAL = 8 |
| 0000FF00 | 008E | 119 | DEF_PROT = ^XFF00 |
| 00000000 | 008E | 120 | LOG_V = 0 |
| 00000001 | 008E | 121 | LOG_M = 1 |
| 00000001 | 008E | 122 | ATTR_V = 1 |
| 00000002 | 008E | 123 | ATTR_M = 2 |
| 00000002 | 008E | 124 | DEF_V = 2 |
| 00000004 | 008E | 125 | DEF_M = 4 |
| 00000004 | 008E | 126 | NAME_ATTR = 4 |
| 00000000 | 008E | 127 | TRAN_ATTR = 0 |
| | 008E | 128 | |


```
008E 130 .SBTTL ALLOCATE DEVICE
008E 131 :+
008E 132 :DCL$ALLOCATE - ALLOCATE DEVICE
008E 133 :
008E 134 :THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE ALLOCATE
008E 135 :COMMAND.
008E 136 :
008E 137 :INPUTS:
008E 138 :
008E 139 :R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
008E 140 :R9 = ADDRESS OF SCRATCH STACK.
008E 141 :R10 = BASE ADDRESS OF COMMAND WORK AREA.
008E 142 :R11 = BASE ADDRESS OF PROCESS WORK AREA.
008E 143 :
008E 144 :OUTPUTS:
008E 145 :
008E 146 :THE SPECIFIED DEVICE IS ALLOCATED AND ASSIGNED THE SPECIFIED LOGICAL
008E 147 :NAME. IF THE LOGICAL NAME WAS PREVIOUSLY ASSIGNED, THEN A MESSAGE TO
008E 148 :THAT EFFECT IS WRITTEN TO THE OUTPUT STREAM.
008E 149 :-
008E 150 :DCL$ALLOCATE:: :ALLOCATE DEVICE
008E 151 :
008E 152 :Allocate and init common logical name data structure.
008E 153 :
59 10 C2 008E 154 :SUBL #16,R9 :ALLOCATE PHYSICAL DEV BUFFER
79 59 D0 0091 155 :MOVL R9,-(R9) :INIT THE EQUIV NAME DESCR
79 10 3C 0094 156 :MOVZWL #16,-(R9) :
79 79 7C 0097 157 :CLRQ -(R9) :ALLOCATE LOG NAME DESCR
55 BE AF 9E 0099 158 :MOVAB LNMS$FILE_DEV,R5 :SET LOGICAL NAME TABLE
54 85 9A 009D 159 :MOVZBL (R5)+,R4 :
79 54 7D 00A0 160 :MOVQ R4,-(R9) :SAVE THE DESCRIPTOR
79 02 D0 00A3 161 :MOVL #PSL$C_SUPER,-(R9) :AND SUPERVISOR MODE
79 01 D0 00A6 162 :MOVL #LOG_M,-(R9) :ASSUME /LOG DEFAULTED
79 79 7C 00A9 163 :CLRQ -(R9) :SET DEFAULT NAME/TRAN ATTRIBUTES
58 59 D0 00AB 164 :MOVL R9,R8 :COPY THE BASE OF THE DATA STRUCTURE
79 D4 00AE 165 :CLRL -(R9) :ALLOCATE /TYPE VALUE
008E 166 :
57 0908 56 D4 00B0 167 :CLRL R6 :SET NO TERMINATOR YET
008E 168 :MOVZWL #SS$_NOSUCHDEV,R7 :PRESET ERROR STATUS
008E 169 :
008E 170 :
008E 171 :Process /log and /type command qualifiers.
008E 172 :
008E 173 :10$: BSBW DCL$GETDVAL :GET NEXT TOKEN
03 55 91 00BA 174 :CMPB R5,#PTR_K_PARAMETR :PARAMETER VALUE?
2C 13 00BD 175 :BEQL 40$ :YES, THEN NO /GENE OR /LOG
FF46' 30 00BF 176 :BSBW DCL$GETNVAL :GET QUALIFIER TYPE
51 00'8F 91 00C2 177 :CMPB #CLISK_ALLO_GENE,R1 :IS IT /GENE
0D 13 00C6 178 :BEQL 20$ :YES, THEN BRANCH
08 A8 01 C8 00C8 179 :BISL #LOG_M,QUAL(R8) :ASSUME /LOG
EB 53 E9 00CC 180 :BLBC R3,10$ :BRANCH IF SO
08 A8 01 CA 00CF 181 :BICL #LOG_M,QUAL(R8) :SET /NOLOG
E2 11 00D3 182 :BRB 10$ :GET NEXT TOKEN
69 D4 00D5 183 :CLRL (R9) :ASSUME /NOGENERIC
DD 53 E8 00D7 184 :BLBS R3,10$ :BRANCH IF SO
69 01 D0 00DA 185 :MOVL #1,(R9) :IT WAS /GENERIC
D8 11 00DD 186 :BRB 10$ :GET NEXT TOKEN
```



```
00DF 187
00DF 188
00DF 189 : Return allocation error.
00DF 190
50 57 D0 00DF 191 90$: MOVL R7,R0 ;SET ERROR STATUS
05 05 00E2 192 RSB ;EXIT
00E3 193
00E3 194
00E3 195 : Process the device names.
00E3 196
05 FF1A' 30 00E3 197 30$: BSBW DCL$GETDVAL ;GET NEXT TOKEN
05 56 91 00E6 198 CMPB R6,#PTR_K_COMMA ;ANOTHER PARAMETER VALUE IN LIST?
03 F4 12 00E9 199 BNEQ 90$ ;NO, THEN ERROR
03 55 91 00EB 200 40$: CMPB R5,#PTR_K_PARAMETR ;PARAMETER VALUE?
03 EF 12 00EE 201 BNEQ 90$ ;NO, THEN ERROR
56 54 D0 00F0 202 MOVL R4,R6 ;SAVE TOKEN TERMINATOR
FF0A' 30 00F3 203 BSBW DCL$COMPSTRING ;REMOVE ANY DOUBLE QUOTES
18 A8 51 7D 00F6 204 MOVQ R1,LOGNAM(R8) ;SAVE THE DESCRIPTOR
FF A241 3A 91 00FA 205 CMPB #^A/:/,-1(R2)[R1] ;DEVICE NAME END WITH A COLON?
03 12 00FF 206 BNEQ 50$ ;IF NEQ NO
18 A8 D7 0101 207 DECL LOGNAM(R8) ;REDUCE LENGTH OF DEVICE NAME
50 69 D0 0104 208 50$: MOVL (R9),R0 ;GET /GENERIC INDICATOR
0107 209 $ALLOC,S LOGNAM(R8),EQUAM(R8),- ;ALLOCATE DEVICE
0107 210 EQUAM(R8),#0,R0
57 50 D0 011B 211 MOVL R0,R7 ;SAVE FINAL STATUS
C2 50 E9 011E 212 BLBC R0,30$ ;IF ERROR, TRY NEXT DEVICE IN LIST
0121 213
0121 214 : Output the device allocated message.
0121 215
0121 216
10 08 A8 00 E1 0121 217 BBC #LOG V,QUAL(R8),80$ ;SKIP IF /NOLOG
20 A8 7F 0126 218 PUSHAQ EQUAM(R8) ;PUSH DESCRIPTOR ADDRESS
51 01 D0 0129 219 MOVL #1,R1 ;SET ARG COUNT
50 0003DDE3 8F D0 012C 220 MOVL #CLIS$ALLOC,R0 ;SET STATUS
FECA' 30 0133 221 BSBW DCL$FORMMSG ;OUTPUT INFORMATIONAL MESSAGE
0136 222
0136 223 : Get the requested logical name.
0136 224
0136 225
03 FEC7' 30 0136 226 80$: BSBW DCL$GETDVAL ;GET NEXT PARAMETER VALUE
03 55 91 0139 227 CMPB R5,#PTR_K_PARAMETR ;PARAMETER VALUE?
02 13 013C 228 BEQL 60$ ;CONTINUE IF LOGICAL NAME FOUND
03 5B 11 013E 229 BRB 95$ ;EXIT IF NOT
05 56 91 0140 230 60$: CMPB R6,#PTR_K_COMMA ;STILL IN P1 LIST?
05 05 12 0143 231 BNEQ 70$ ;IF P2 FOUND, ASSIGN THE LOGICAL NAM
56 54 D0 0145 232 MOVL R4,R6 ;COPY TERMINATOR TYPE CODE
EC 11 0148 233 BRB 80$ ;LOOP UNTIL P2 OR EOL FOUND
014A 234
FF A241 FEB3' 30 014A 235 70$: BSBW DCL$COMPSTRING ;REMOVE QUOTATION MARKS
3A 91 014D 236 CMPB #^A/:/,-1(R2)[R1] ;LOGICAL NAME END WITH COLON?
02 12 0152 237 BNEQ 75$ ;IF NEQ NO
51 D7 0154 238 DECL R1 ;REDUCE LENGTH OF LOGICAL NAME
18 A8 51 7D 0156 239 75$: MOVQ R1,LOGNAM(R8) ;SAVE THE LOGICAL NAME
015A 240
015A 241 : Create the required item list.
015A 242
015A 243
```

| | | | | | | | | |
|----|----------|----|----|------|------|-----------|----------------------|------------------------------------|
| 7E | 20 | 7E | 7C | 015A | 244 | CLRQ | -(SP) | ;TERMINATE THE LIST, ZERO LEN ADDR |
| 02 | AE | 02 | 7D | 015C | 245 | MOVQ | EQUAM(R8),-(SP) | ;SET THE EQUIV NAME DESCR |
| 57 | 5E | 02 | 80 | 0160 | 246 | MOVW | #LNMS_STRING,2(SP) | ;SET THE ITEM TYPE |
| | | 5E | D0 | 0164 | 247 | MOVL | SP,R7 | ;GET THE ITEM LIST ADDRESS |
| | | | | 0167 | 248 | | | |
| | | | | 0167 | 249 | \$CRELM_S | ATTR=NAME ATTR(R8),- | ;CREATE THE REQUESTED NAME |
| | | | | 0167 | 250 | | TABNAM=TABNAM(R8),- | |
| | | | | 0167 | 251 | | LOGNAM=LOGNAM(R8),- | |
| | | | | 0167 | 252 | | ACMODE=ACMODE(R8),- | |
| | | | | 0167 | 253 | | ITMLST=(R7) | |
| | | | | 017C | 254 | | | |
| 5E | 10 | | C0 | 017C | 255 | ADDL | #4*4,SP | ;POP THE ITEM LIST |
| | | | | 017F | 256 | | | |
| | | | | 017F | 257 | | | |
| | | | | 017F | 258 | | | |
| | | | | 017F | 259 | | | |
| 50 | 0631 | 8F | B1 | 017F | 260 | | | |
| | | 1C | 12 | 0184 | 261 | CMPW | #SS\$_SUPERSEDE,R0 | ;PREVIOUS ASSIGNMENT SUPERSEDED? |
| 10 | 08 | A8 | 00 | E1 | 0186 | BNEQ | 96\$ | ;IF NEQ NO |
| | 18 | A8 | 9F | 018B | 262 | BBC | #LOG_V,QUAL(R8),95\$ | ;BRANCH IF /NOLOG |
| | 51 | 01 | D0 | 018E | 263 | PUSHAB | LOGNAM(R8) | ;SET LOGICAL NAME ADDRESS |
| 50 | 0003DDEB | 8F | D0 | 0191 | 264 | MOVL | #1,R1 | ;SET FAO COUNT |
| | FE65 | | 30 | 0198 | 265 | MOVL | #CLIS_SUPERSEDE,R0 | ;SET STATUS |
| | | | | 019B | 266 | BSBW | DCL\$FORMMSG | ;OUTPUT MESSAGE |
| | | | | 019B | 267 | | | |
| | | | | 019B | 268 | 95\$: | STATUS NORMAL | ;RETURN SUCCESS |
| | | | 05 | 01A2 | 269 | 96\$: | RSB | ;EXIT |

: Output informational message if appropriate.

```
01A3 271 .SBTTL ASSIGN LOGICAL NAME TO EQUIVALENCE STRING
01A3 272
01A3 273 :+ DCL$ASSIGN - ASSIGN LOGICAL NAME TO EQUIVALENCE STRING
01A3 274
01A3 275 THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE ASSIGN
01A3 276 COMMAND.
01A3 277
01A3 278 INPUTS:
01A3 279
01A3 280 R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
01A3 281 R9 = ADDRESS OF SCRATCH STACK.
01A3 282 R10 = BASE ADDRESS OF COMMAND WORK AREA.
01A3 283 R11 = BASE ADDRESS OF PROCESS WORK AREA.
01A3 284
01A3 285 OUTPUTS:
01A3 286
01A3 287 THE SPECIFIED LOGICAL NAME IS ASSIGNED TO THE SPECIFIED EQUIVALENCE
01A3 288 STRING. IF A PREVIOUS LOGICAL ASSIGNMENT IS SUPERSEDED, THEN A
01A3 289 MESSAGE TO THAT EFFECT IS WRITTEN TO THE OUTPUT STREAM.
01A3 290
01A3 291
01A3 292 DCL$ASSIGN:: :ASSIGN LOGICAL NAME TO EQUIVALENCE
01A3 293
01A3 294 Parse the common qualifiers and the logical name string.
01A3 295
01A3 296 BSBW COMMON_QUAL :PROCESS COMMON QUALIFIERS
01A3 297 CLRQ -(R9) :ALLOCATE SPACE FOR EQUIV NAME
01A3 298 MOVQ LOGNAM(R8),R1 :GET EQUIV NAME DESCR
01A3 299 BSBW DCL$COMPRESS :COMPRESS THE STRING
01A3 300 MOVQ R1,EQUAM(R8) :SAVE EQUIV NAME
01B3 301
01B3 302
01B3 303 Init the item list. Insert the default translation attributes.
01B3 304
01B3 305 MOVAB -64*6+4-4(SP),SP :ALLOCATE ROOM FOR A 128 ITEM LIST
01B3 306 MOVL SP,R7 :SAVE THE ADDRESS OF THE LIST
01B3 307
01B3 308 MOVL #LNMS_ATTRIBUTES@16+4,(R7)+ :SET THE ITEM TYPE
01B3 309 MOVAL TRAN_ATTR(R8),(R7)+ :SET THE DEFAULT ATTRIBUTES ADDR
01B3 310 CLRL (R7)+ :ZERO THE RETURN LENGTH ADDR
01B3 311 MOVL #1,R6 :MARK DEFAULT ATTRIBUTES SET
01CA 312
01CA 313
01CA 314 Loop getting equivalence strings and their attributes.
01CA 315 Build the item list.
01CA 316
01CA 317 25$: BSBW GET_TRAN_ATTR :CHECK FOR NEW TRAN ATTRIBUTES
01CA 318 BLBC R0,30$ :BRANCH IF NO LOCAL QUALIFIER
01CA 319 MOVL #LNMS_ATTRIBUTES@16+4,(R7)+ :SET THE ITEM TYPE
01CA 320 MOVL R3,-(R9) :SAVE THE ATTRIBUTES
01CA 321 MOVL R9,(R7)+ :SET THE ATTRIBUTES ADDR
01CA 322 CLRL (R7)+ :ZERO THE RETURN LENGTH ADDR
01CA 323 CLRL R6 :MARK NEW ATTRIBUTES SET
01CA 324 BRB 40$ :PROCESS THE PARAMETER
01CA 325 30$: BLBS R6,40$ :SKIP IF DEFAULTS IN EFFECT
01CA 326 MOVL #LNMS_ATTRIBUTES@16+4,(R7)+ :SET THE ITEM TYPE
01CA 327 MOVAL TRAN_ATTR(R8),(R7)+ :SET THE DEFAULT ATTRIBUTES ADDR
```

0221 30
79 7C
51 18 A8 7D
FE51' 30
20 A8 51 7D

SE F9FC CE 9E
57 5E DO

87 00030004 8F DO
87 68 DE
87 87 D4
56 01 DO

D377 30
13 50 E9
87 00030004 8F DO
79 53 DO
87 59 DO
87 87 D4
56 87 D4
12 11
OF 56 E8
87 00030004 8F DO
87 68 DE


```
      56 87 D4 01F0 328      CLRL (R7)+      ;ZERO THE RETURN LENGTH ADDR
      01 D0 01F2 329      MOVL #1,R6      ;MARK NEW ATTRIBUTES SET
      01F5 330
87 20 A8 7D 01F5 331 40$: MOVQ EQUAM(R8),(R7)+      ;SAVE THE EQUIV NAME DESCR
FA A7 02 B0 01F9 332      MOVW #LNMS_STRING,-6(R7)      ;SET THE ITEM TYPE
      87 D4 01FD 333      CLRL (R7)+      ;ZERO THE RETURN LENGTH ADDR
      0C C3 01FF 334      SUBL3 #PTR_K_LENGTH,-      ;GET ADDRESS OF TOKEN DESCRIPTOR
50 BA AA 01 0201 335      WRK_C_RSLNXT(R10),R0
02 06 A0 91 0204 336      CMPB PTR_B_PARMCNT(R0),#2      ;HAVE WE FOUND THE LOGICAL NAME?
      09 13 0208 337      BEQL 43$      ;YES, THEN TERMINATE ITEM LIST
      FDF3' 30 020A 338      BSBW DCL$COMPRESS      ;COMPRESS THE STRING
20 A8 51 7D 020D 339      MOVQ R1,EQUAM(R8)      ;STORE THE LATEST STRING DESCR
      87 11 0211 340      BRB 25$      ;CHECK FOR NEW ATTRIBUTES
      87 D4 0213 341
      0215 342 43$: CLRL (R7)+      ;TERMINATE THE LIST
      0215 343
      0215 344
      0215 345      ; Process logical name string.
      0215 346
      FDE8' 30 0215 347      BSBW DCL$COMPSTRING      ;REMOVE QUOTATION MARKS
FF A241 3A 91 0218 348      CMPB #A/:/, -1(R2)[R1]      ;LOGICAL NAME END WITH COLON?
      02 12 021D 349      BNEQ 45$      ;IF NEQ NO
      51 D7 021F 350      DECL R1      ;REDUCE LENGTH OF LOGICAL NAME
18 A8 51 7D 0221 351 45$: MOVQ R1,LOGNAM(R8)      ;SAVE LOGICAL NAME DESCRIPTOR
      031C 30 0225 352      BSBW GET_TRAN_ATTR      ;CHECK FOR NEW TRAN ATTRIBUTES
      03 50 E9 0228 353      BLBC R0,47$      ;BRANCH IF NO LOCAL QUALIFIER
68 53 D0 022B 354      MOVL R3,TRAN_ATTR(R8)      ;SAVE THE ATTRIBUTES
      007C 31 022E 355 47$: BRW COMMON_CRELNM      ;CREATE THE LOGICAL NAME
```

```
0231 357 .SBTTL DEFINE LOGICAL NAME EQUIVALENCE
0231 358 :+
0231 359 DCL$DEFINE - DEFINE LOGICAL NAME EQUIVALENCE
0231 360
0231 361 THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE DEFINE
0231 362 COMMAND.
0231 363
0231 364 INPUTS:
0231 365
0231 366 R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
0231 367 R9 = ADDRESS OF SCRATCH STACK.
0231 368 R10 = BASE ADDRESS OF COMMAND WORK AREA.
0231 369 R11 = BASE ADDRESS OF PROCESS WORK AREA.
0231 370
0231 371 OUTPUTS:
0231 372
0231 373 THE SPECIFIED LOGICAL NAME IS ASSIGNED TO THE SPECIFIED EQUIVALENCE
0231 374 STRING. IF A PREVIOUS LOGICAL ASSIGNMENT IS SUPERSEDED, THEN A
0231 375 MESSAGE TO THAT EFFECT IS WRITTEN TO THE OUTPUT STREAM.
0231 376
0231 377
0231 378 DCL$DEFINE:: :DEFINE LOGICAL NAME EQUIVALENCE
0231 379
0231 380 Parse the common qualifiers and the logical name string.
0231 381
0231 382 BSBW COMMON_QUAL :PROCESS COMMON QUALIFIERS
0231 383 CLRQ -(R9) :ALLOCATE SPACE FOR EQUIVALENCE NAME
0231 384 MOVQ LOGNAM(R8),R1 :GET LOGICAL NAME DESCR
0231 385 BSBW DCL$COMPSTRING :REMOVE QUOTES FROM LOGICAL NAME
0231 386 MOVQ R1,LOGNAM(R8) :SAVE LOGICAL NAME
0231 387
0231 388
0231 389 Init the item list. Insert the default translation attributes.
0231 390
0231 391 MOVAB -64*6*4-4(SP),SP :ALLOCATE ROOM FOR A 128 ITEM LIST
0231 392 MOVL SP,R7 :SAVE THE ADDRESS OF THE LIST
0231 393
0231 394 MOVL #LNMS_ATTRIBUTES@16+4,(R7)+ :SET THE ITEM TYPE
0231 395 MOVAL TRAN_ATTR(R8),(R7)+ :SET THE DEFAULT ATTRIBUTES ADDR
0231 396 CLRL (R7)+ :ZERO THE RETURN LENGTH ADDR
0231 397
0231 398 BSBW GET_TRAN_ATTR :CHECK FOR NEW TRAN ATTRIBUTES
0231 399 BLBC R0,23$ :BRANCH IF NO LOCAL QUALIFIER
0231 400 MOVL R3,TRAN_ATTR(R8) :SAVE THE ATTRIBUTES
0231 401 BSBW DCL$COMPRESS :COMPRESS THE STRING
0231 402 MOVQ R1,EQUAM(R8) :LOAD THE PIPELINE
0231 403 MOVL #1,R6 :MARK DEFAULT ATTRIBUTES SET
0231 404
0231 405
0231 406 Loop getting equivalence strings and their attributes.
0231 407 Build the item list.
0231 408
0231 409 25$: BSBW GET_TRAN_ATTR :CHECK FOR NEW TRAN ATTRIBUTES
0231 410 BLBC R0,30$ :BRANCH IF NO LOCAL QUALIFIER
0231 411 MOVL #LNMS_ATTRIBUTES@16+4,(R7)+ :SET THE ITEM TYPE
0231 412 MOVL R3,-(R9) :SAVE THE ATTRIBUTES
0231 413 MOVL R9,(R7)+ :SET THE ATTRIBUTES ADDR
```

0193 30 0231 382
79 7C 0234 383
51 18 A8 7D 0236 384
FDC3' 30 023A 385
18 A8 51 7D 023D 386
0241 387
0241 388
0241 389
0241 390
5E F9FC CE 9E 0241 391
57 5E D0 0246 392
0249 393
87 00030004 8F D0 0249 394
87 68 DE 0250 395
87 D4 0253 396
0255 397
02EC 30 0255 398
03 50 E9 0258 399
68 53 D0 025B 400
FD9F' 30 025E 401 23\$: BSBW
20 A8 51 7D 0261 402 MOVQ
56 01 D0 0265 403 MOVL
0268 404
0268 405
0268 406
0268 407
0268 408
02D9 30 0268 409 25\$: BSBW
13 50 E9 026B 410 BLBC
87 00030004 8F D0 026E 411 MOVL
79 53 D0 0275 412 MOVL
87 59 D0 0278 413 MOVL

| | | | | | | | | | | | |
|----|-------|----------|-------|------|------|-------|-----|-------|-------|----------------------------|----------------------------------|
| 87 | | 00030004 | OF 56 | 8F | D0 | 0284 | 418 | 30\$: | MOV | LNMS_ATTRIBUTES@16+4,(R7)+ | :ZERO THE RETURN LENGTH ADDR |
| | | | 87 68 | DE | 0288 | 419 | | | MOVAL | TRAN_ATTR(R8),(R7)+ | :MARK NEW ATTRIBUTES SET |
| | | | 56 01 | D0 | 0290 | 421 | | | CLRL | (R7)+ | :PROCESS THE PARAMETER |
| | | | | | 0293 | 422 | | | MOV | #1,R6 | :SKIP IF DEFAULTS IN EFFECT |
| 87 | 20 | A8 | 7D | 0293 | 423 | 40\$: | | | MOVQ | EQUAM(R8),(R7)+ | :SET THE ITEM TYPE |
| FA | A7 | 02 | B0 | 0297 | 424 | | | | MOVW | LNMS_STRING,-6(R7) | :SET THE DEFAULT ATTRIBUTES ADDR |
| | | 87 | D4 | 029B | 425 | | | | CLRL | (R7)+ | :ZERO THE RETURN LENGTH ADDR |
| 55 | 04 | 91 | 029D | 426 | | | | | CMPB | PTR_K_ENDLINE,R5 | :EOL? |
| | 09 | 13 | 02A0 | 427 | | | | | BEQL | 43\$ | :YES, THEN TERMINATE ITEM LIST |
| | FD5B' | 30 | 02A2 | 428 | | | | | BSBW | DCL\$COMPRESS | :COMPRESS THE STRING |
| 20 | A8 | 51 | 7D | 02A5 | 429 | | | | MOVQ | R1,EQUAM(R8) | :STORE THE LATEST STRING DESCR |
| | | BD | 11 | 02A9 | 430 | | | | BRB | 25\$ | :CHECK FOR NEW ATTRIBUTES |
| | | | | 02AB | 431 | | | | | | |
| | | 87 | D4 | 02AB | 432 | 43\$: | | | CLRL | (R7)+ | :TERMINATE THE LIST |
| | | | | 02AD | 433 | | | | | | |


```
02AD 435 :  
02AD 436 : Check for SYS$OUTPUT. Do special processing if appropriate.  
02AD 437 :  
02AD 438 COMMON_CRELNM:  
57 1C C2 02AD 439      SUBL      #6*4+4,R7      ;WAS MORE THAN ONE VALUE SUPPLIED  
5E 57 D1 02B0 440      CMPL      R7,SP      :  
      05 12 02B3 441      BNEQ      45$      :YES, THEN BRANCH  
      00D9 30 02B5 442      BSBW      TESTOUT      :IS LOGICAL NAME SYS$OUTPUT?  
      46 13 02B8 443      BEQL      80$      :YES, THEN BRANCH  
      02BA 444 :  
      02BA 445 :  
      02BA 446 : Create the requested logical names.  
      02BA 447 :  
00 08 A8 01 E1 02BA 448 45$: BBC      #ATTR V,QUAL(R8),47$      ;BRANCH IF QUALIFIER NOT SEEN  
      57 5E D0 02BF 449      BICL      #LNMSM_CRELOG,NAME_ATTR(R8) ;DISABLE CRELOG ATTRIBUTE  
      02C2 451 47$: MOVL      SP,R7      ;GET THE ITEM LIST ADDRESS  
      02C2 452      $CRELNM,S ATTR=NAME ATTR(R8),- ;CREATE THE REQUESTED NAME  
      02C2 453      TABNAM=TABNAM(R8),- ;  
      02C2 454      LOGNAM=LOGNAM(R8),- ;  
      02C2 455      ACMODE=ACMODE(R8),- ;  
      02D7 456      ITMLST=(R7) ;  
      02D7 457 :  
      02D7 458 : Output informational message if appropriate.  
      02D7 459 :  
50 0631 8F B1 02D7 460      CMPW      #$$$_SUPERSEDE,R0      ;PREVIOUS ASSIGNMENT SUPERSEDED?  
      1C 12 02DC 461      BNEQ      60$      ;IF NEQ NO  
10 08 A8 00 E1 02DE 462      BBC      #LOG V,QUAL(R8),50$      ;BRANCH IF /NOLOG  
      18 A8 9F 02E3 463      PUSHAB   LOGNAM(R8)      ;SET LOGICAL NAME ADDRESS  
      51 01 D0 02E6 464      MOVL      #1,R1      ;SET FAO COUNT  
50 0003DDEB 8F D0 02E9 465      MOVL      #CLIS_SUPERSEDE,R0      ;SET STATUS  
      FDD0' 30 02F0 466      BSBW      DCL$FORMMSG      ;OUTPUT MESSAGE  
      02F3 467 :  
      02F3 468 50$: STATUS      NORMAL      ;SET NORMAL COMPLETION  
5E 0604 CE 9E 02FA 469 60$: MOVAB   64*6*4+4(SP),SP      ;RESTORE THE STACK  
      05 02FF 470      RSB      :  
      0300 471 :  
      0300 472 :  
      0300 473 : Update SYS$OUTPUT.  
      0300 474 :  
51 20 A8 7D 0300 475 80$: MOVQ      EQUAM(R8),R1      ;GET DESCRIPTOR OF EQUIVALENCE NAME  
      FCF9' 30 0304 476      BSBW      DCL$OPEN_OUTPUT      ;OPEN SPECIFIED OUTPUT FILE  
      FO 50 E9 0307 477      BLBC      R0,60$      ;LEAVE EVERYTHING ALONE IF ERROR  
58 00BC CB D0 030A 478      MOVL      PRCL IDFLNK(R11),R8      ;POINT TO THE SYS$OUTPUT INFORMATION  
      FCEE' 30 030F 479      BSBW      DCL$CREATE_OUTPUT      ;CREATE THE SYS$OUTPUT LOGICAL NAME  
      DF 11 0312 480      BRB      50$      ;
```

```
0314 482 .SBTTL DEALLOCATE DEVICE
0314 483
0314 484 :+ DCL$DEALLOCAT - DEALLOCATE DEVICE
0314 485
0314 486 : THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE DEALLOCATE
0314 487 : DCL$ COMMAND.
0314 488
0314 489 : INPUTS:
0314 490
0314 491 : R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
0314 492 : R9 = ADDRESS OF SCRATCH STACK.
0314 493 : R10 = BASE ADDRESS OF COMMAND WORK AREA.
0314 494 : R11 = BASE ADDRESS OF PROCESS WORK AREA.
0314 495
0314 496 : OUTPUTS:
0314 497
0314 498 : THE SPECIFIED DEVICE IS DEALLOCATED OR ALL DEVICES ARE DEALLOCATED.
0314 499 :-
0314 500
0314 501 DCL$DEALLOCAT::
0314 502 BSBW DCL$GETDVAL ;DEALLOCATE DEVICE
0314 503 CMPB #PTR_K_PARAMETR,R5 ;GET TOKEN DESCRIPTOR
0314 504 BEQL 10$ ;ITEM TYPE PARAMETER?
0314 505 CLRL R9 ;YES, PROCESS IT
0314 506 BRB 90$ ;NO, ASSUME /ALL
0314 507 10$: BSBW DCL$COMPSTRING ;DEALLOCATE THEM ALL
0314 508 MOVQ R1,-(R9) ;REMOVE EXTERNAL QUOTATION MARKS
0314 509 CMPB #^A/:/,-1(R2)[R1] ;SAVE LOGICAL NAME
0314 510 BNEQ 90$ ;STRING END WITH A COLON
0314 511 DECL (R9) ;BR IF NO
0314 512 90$: $DALLOC_S (R9) ;REMOVE COLON FROM STRING
0314 513 RSB ;DEALLOCATE DEVICE
0314 514 :
```

55 FCE9' 30 0314 502
03 91 0317 503
04 13 031A 504
59 D4 031C 505
OF 11 031E 506
FCDD' 30 0320 507
79 51 7D 0323 508
FF A241 3A 91 0326 509
02 12 032B 510
69 D7 032D 511
05 032F 512
033A 513

```
033B 515 .SBTTL DEASSIGN LOGICAL NAME EQUIVALENCE
033B 516
033B 517 :+ DCL$DEASSIGN - DEASSIGN LOGICAL NAME EQUIVALENCE
033B 518
033B 519 THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE DEASSIGN DCL$
033B 520 COMMAND.
033B 521
033B 522 INPUTS:
033B 523
033B 524 R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
033B 525 R9 = ADDRESS OF SCRATCH STACK.
033B 526 R10 = BASE ADDRESS OF COMMAND WORK AREA.
033B 527 R11 = BASE ADDRESS OF PROCESS WORK AREA.
033B 528
033B 529 OUTPUTS:
033B 530
033B 531 THE SPECIFIED LOGICAL NAME EQUIVALENCE OR ALL LOGICAL NAME EQUIVALENCES
033B 532 ARE DEASSIGNED.
033B 533 :-
033B 534
033B 535 DCL$DEASSIGN::
033B 536 BSBW COMMON_QUAL :DEASSIGN LOGICAL NAME EQUIVALENCE
033E 537 CLRL R7 :PROCESS COMMON QUALIFIERS
0340 538 BBC #DEF_V,QUAL(R8),3$ :ASSUME DOING /ALL
0345 539 BBS #LOG_V,QUAL(R8),5$ :SKIP IF DEFAULTED
034A 540 3$: MOVQ LOGNAM(R8),R1 :BR IF DOING /ALL
034E 541 BSBW DCL$COMPSTRING :GET LOGICAL NAME
0351 542 MOVQ R1,LOGNAM(R8) :REMOVE EXTERNAL QUOTATION MARKS
0355 543 MOVAB LOGNAM(R8),R7 :SAVE LOGICAL NAME
0359 544 CMPB #^A/:/, -1(R2)[R1] :COPY THE DESCRIPTOR ADDRESS
035E 545 BNEQ 5$ :STRING END WITH A COLON
0360 546 DECL LOGNAM(R8) :BR IF NO
0363 547 5$: BSBB TESTOUT :REMOVE COLON FROM STRING
0365 548 BEQL 10$ :IS LOGICAL NAME SYSS$OUTPUT?
0367 549 $DELLNM,S TABNAM=TABNAM(R8),- :YES, THEN SKIP
0367 550 LOGNAM=(R7),- :DEASSIGN LOGICAL NAME EQUIVALENCE
0367 551 ACMODE=ACMODE(R8)
0376 552 TSTL R7 :DEASSIGN/ALL?
0378 553 BEQL 20$ :YES, THEN RECREATE SYSS$OUTPUT
037A 554 RSB
037B 555
037B 556 10$: MOVAB PRC_W_OUTIFI(R11),R2 :GET ADDRESS OF SYSS$OUTPUT INFORMATION
0380 557 MOVL PRC_L_IDFLNK(R11),R8 :GET ADDRESS OF CURRENT IDF BLOCK
0385 558 BSBW DCL$RESTORE_OUTPUT :RESTORE PROCESS PERMANENT SYSS$OUTPUT
0388 559 20$: MOVL PRC_L_IDFLNK(R11),R8 :GET ADDRESS OF CURRENT IDF BLOCK
038D 560 BSBW DCL$CREATE_OUTPUT :CREATE SYSS$OUTPUT LOGICAL NAME
0390 561 RSB
```

| | | | | | | | | |
|----|------|------|------|----|------|-----|-------|--|
| 05 | 08 | A8 | 02 | D4 | 033E | 537 | | |
| 19 | 08 | A8 | 00 | E0 | 0345 | 539 | | |
| | 51 | 18 | A8 | 7D | 034A | 540 | 3\$: | |
| | | | FCAF | 30 | 034E | 541 | | |
| | 18 | A8 | 51 | 7D | 0351 | 542 | | |
| | 57 | 18 | A8 | 7E | 0355 | 543 | | |
| FF | A241 | | 3A | 91 | 0359 | 544 | | |
| | | | 03 | 12 | 035E | 545 | | |
| | | 18 | A8 | D7 | 0360 | 546 | | |
| | | | 2C | 10 | 0363 | 547 | 5\$: | |
| | | | 14 | 13 | 0365 | 548 | | |
| | | | | | 0367 | 549 | | |
| | | | | | 0367 | 550 | | |
| | | | | | 0367 | 551 | | |
| | | 57 | | D5 | 0376 | 552 | | |
| | | 0E | | 13 | 0378 | 553 | | |
| | | | | 05 | 037A | 554 | | |
| | | | | | 037B | 555 | | |
| 52 | 0114 | CB | | 9E | 037B | 556 | 10\$: | |
| 58 | 00BC | CB | | D0 | 0380 | 557 | | |
| | | FC78 | | 30 | 0385 | 558 | | |
| 58 | 00BC | CB | | D0 | 0388 | 559 | 20\$: | |
| | | FC70 | | 30 | 038D | 560 | | |
| | | | | 05 | 0390 | 561 | | |


```
0391 563 .SBTTL TEST IF LOGICAL NAME IS SYSS$OUTPUT
0391 564 :
0391 565 : SUBROUTINE TO TEST IF LOGICAL NAME IS SYSS$OUTPUT
0391 566 :
0391 567 : ON OUTPUT, 'Z'= 1 IF SYSS$OUTPUT IS SPECIFIED
0391 568 :
0391 569 : CLOBBERS R0-R3
0391 570 :
0391 571 TESTOUT:
0391 572 TSTL LOGNAM(R8) ;WAS A LOGICAL NAME SPECIFIED?
0394 573 BEQL 20$ ;RETURN IF NOT
0396 574 BBS #ATTR_V,QUAL(R8),20$ ;BRANCH IF ATTRIBUTES SPECIFIED
0398 575 CMPL #PSL$C_USER,ACMODE(R8) ;IS LOGICAL NAME USER MODE?
039F 576 BEQL 20$ ;YES, THEN RETURN
03A1 577 CMPB LNM$PROCESS,TABNAM(R8) ;COMPARE LENGTH OF TABLE NAME
03A7 578 BNEQ 20$ ;RETURN IF NOT EQUAL
03A9 579 CMPC TABNAM(R8),@TABNAM+4(R8) ;COMPARE ACTUAL STRING
03AE 580 LNM$PROCESS+1
03B1 581 BNEQ 20$ ;RETURN IF NOT EQUAL
03B3 582 CMPB OUTPUTNAM,LOGNAM(R8) ;COMPARE LENGTH OF OUTPUT
03B9 583 BNEQ 20$ ;RETURN IF NOT EQUAL
03BB 584 CMPC LOGNAM(R8),@LOGNAM+4(R8) ;COMPARE ACTUAL STRING
03C0 585 OUTPUTNAM+1
03C3 586 10$: RSB
03C4 587
03C4 588 20$: TSTL #1 ;SET FAILURE STATUS
03C6 589 RSB
```

| | | | |
|----------|----|------|-----|
| 18 A8 | D5 | 0391 | 572 |
| 29 08 A8 | 13 | 0394 | 573 |
| OC A8 | E0 | 0396 | 574 |
| | D1 | 0398 | 575 |
| | 13 | 039F | 576 |
| 10 A8 | 91 | 03A1 | 577 |
| | 12 | 03A7 | 578 |
| 14 B8 | 29 | 03A9 | 579 |
| | | 03AE | 580 |
| | 11 | 03B1 | 581 |
| 18 A8 | 91 | 03B3 | 582 |
| | 12 | 03B9 | 583 |
| 1C B8 | 29 | 03BB | 584 |
| | | 03C0 | 585 |
| | 05 | 03C3 | 586 |
| | | 03C4 | 587 |
| 01 | D5 | 03C4 | 588 |
| | 05 | 03C6 | 589 |

```

03C7 591 .SBTTL PROCESS COMMON COMMAND QUALIFIERS
03C7 592
03C7 593 SUBROUTINE TO PROCESS COMMON COMMAND QUALIFIERS
03C7 594
03C7 595 ON INPUT, R9 = ADDRESS OF SCRATCH STACK
03C7 596
03C7 597 ON OUTPUT, SCRATCH STACK LOOKS LIKE
03C7 598
03C7 599
03C7 600
03C7 601
03C7 602
03C7 603
03C7 604
03C7 605
03C7 606
03C7 607
03C7 608
03C7 609
03C7 610
03C7 611
03C7 612
03C7 613
03C7 614
03C7 615
03C7 616
03C7 617
03C7 618
03C7 619
03C7 620
03C7 621
03C7 622
03C7 623
03C7 624
03C7 625
03C7 626
03C7 627
03C7 628
03C7 629
03C7 630
03C7 631
03C7 632
03C7 633
03C7 634
03C7 635
03C7 636
03C7 637
03C7 638
03C7 639
03C7 640
03C7 641
03C7 642
03C7 643
03C7 644
03C7 645
03C7 646
03C7 647
03C7 648
03C7 649
03C7 650
03C7 651
03C7 652
03C7 653
03C7 654
03C7 655
03C7 656
03C7 657
03C7 658
03C7 659
03C7 660
03C7 661
03C7 662
03C7 663
03C7 664
03C7 665
03C7 666
03C7 667
03C7 668
03C7 669
03C7 670
03C7 671
03C7 672
03C7 673
03C7 674
03C7 675
03C7 676
03C7 677
03C7 678
03C7 679
03C7 680
03C7 681
03C7 682
03C7 683
03C7 684
03C7 685
03C7 686
03C7 687
03C7 688
03C7 689
03C7 690
03C7 691
03C7 692
03C7 693
03C7 694
03C7 695
03C7 696
03C7 697
03C7 698
03C7 699
03C7 700
03C7 701
03C7 702
03C7 703
03C7 704
03C7 705
03C7 706
03C7 707
03C7 708
03C7 709
03C7 710
03C7 711
03C7 712
03C7 713
03C7 714
03C7 715
03C7 716
03C7 717
03C7 718
03C7 719
03C7 720
03C7 721
03C7 722
03C7 723
03C7 724
03C7 725
03C7 726
03C7 727
03C7 728
03C7 729
03C7 730
03C7 731
03C7 732
03C7 733
03C7 734
03C7 735
03C7 736
03C7 737
03C7 738
03C7 739
03C7 740
03C7 741
03C7 742
03C7 743
03C7 744
03C7 745
03C7 746
03C7 747
03C7 748
03C7 749
03C7 750
03C7 751
03C7 752
03C7 753
03C7 754
03C7 755
03C7 756
03C7 757
03C7 758
03C7 759
03C7 760
03C7 761
03C7 762
03C7 763
03C7 764
03C7 765
03C7 766
03C7 767
03C7 768
03C7 769
03C7 770
03C7 771
03C7 772
03C7 773
03C7 774
03C7 775
03C7 776
03C7 777
03C7 778
03C7 779
03C7 780
03C7 781
03C7 782
03C7 783
03C7 784
03C7 785
03C7 786
03C7 787
03C7 788
03C7 789
03C7 790
03C7 791
03C7 792
03C7 793
03C7 794
03C7 795
03C7 796
03C7 797
03C7 798
03C7 799
03C7 800
03C7 801
03C7 802
03C7 803
03C7 804
03C7 805
03C7 806
03C7 807
03C7 808
03C7 809
03C7 810
03C7 811
03C7 812
03C7 813
03C7 814
03C7 815
03C7 816
03C7 817
03C7 818
03C7 819
03C7 820
03C7 821
03C7 822
03C7 823
03C7 824
03C7 825
03C7 826
03C7 827
03C7 828
03C7 829
03C7 830
03C7 831
03C7 832
03C7 833
03C7 834
03C7 835
03C7 836
03C7 837
03C7 838
03C7 839
03C7 840
03C7 841
03C7 842
03C7 843
03C7 844
03C7 845
03C7 846
03C7 847
03C7 848
03C7 849
03C7 850
03C7 851
03C7 852
03C7 853
03C7 854
03C7 855
03C7 856
03C7 857
03C7 858
03C7 859
03C7 860
03C7 861
03C7 862
03C7 863
03C7 864
03C7 865
03C7 866
03C7 867
03C7 868
03C7 869
03C7 870
03C7 871
03C7 872
03C7 873
03C7 874
03C7 875
03C7 876
03C7 877
03C7 878
03C7 879
03C7 880
03C7 881
03C7 882
03C7 883
03C7 884
03C7 885
03C7 886
03C7 887
03C7 888
03C7 889
03C7 890
03C7 891
03C7 892
03C7 893
03C7 894
03C7 895
03C7 896
03C7 897
03C7 898
03C7 899
03C7 900
03C7 901
03C7 902
03C7 903
03C7 904
03C7 905
03C7 906
03C7 907
03C7 908
03C7 909
03C7 910
03C7 911
03C7 912
03C7 913
03C7 914
03C7 915
03C7 916
03C7 917
03C7 918
03C7 919
03C7 920
03C7 921
03C7 922
03C7 923
03C7 924
03C7 925
03C7 926
03C7 927
03C7 928
03C7 929
03C7 930
03C7 931
03C7 932
03C7 933
03C7 934
03C7 935
03C7 936
03C7 937
03C7 938
03C7 939
03C7 940
03C7 941
03C7 942
03C7 943
03C7 944
03C7 945
03C7 946
03C7 947
03C7 948
03C7 949
03C7 950
03C7 951
03C7 952
03C7 953
03C7 954
03C7 955
03C7 956
03C7 957
03C7 958
03C7 959
03C7 960
03C7 961
03C7 962
03C7 963
03C7 964
03C7 965
03C7 966
03C7 967
03C7 968
03C7 969
03C7 970
03C7 971
03C7 972
03C7 973
03C7 974
03C7 975
03C7 976
03C7 977
03C7 978
03C7 979
03C7 980
03C7 981
03C7 982
03C7 983
03C7 984
03C7 985
03C7 986
03C7 987
03C7 988
03C7 989
03C7 990
03C7 991
03C7 992
03C7 993
03C7 994
03C7 995
03C7 996
03C7 997
03C7 998
03C7 999
03C7 1000
03C7 1001
03C7 1002
03C7 1003
03C7 1004
03C7 1005
03C7 1006
03C7 1007
03C7 1008
03C7 1009
03C7 1010
03C7 1011
03C7 1012
03C7 1013
03C7 1014
03C7 1015
03C7 1016
03C7 1017
03C7 1018
03C7 1019
03C7 1020
03C7 1021
03C7 1022
03C7 1023
03C7 1024
03C7 1025
03C7 1026
03C7 1027
03C7 1028
03C7 1029
03C7 1030
03C7 1031
03C7 1032
03C7 1033
03C7 1034
03C7 1035
03C7 1036
03
```

| | | | | | | | |
|-------------|------|----|------|------|--------|------------------------------------|--------------------------------------|
| 00000000'8F | 51 | D1 | 042A | 648 | CMPL | R1,#CLISK_DEFI_SYST | :QUALIFIER MATCH? |
| | 7F | 13 | 0431 | 649 | BEQL | 150\$ | :YES, THEN BRANCH |
| 00000000'8F | 51 | D1 | 0433 | 650 | CMPL | R1,#CLISK_DEFI_TABL | :QUALIFIER MATCH? |
| | 03 | 12 | 043A | 651 | BNEQ | 60\$ | :NO, CHECK NEXT |
| | 008A | 31 | 043C | 652 | BRW | 160\$ | :YES, THEN BRANCH |
| 00000000'8F | 51 | D1 | 043F | 653 | CMPL | R1,#CLISK_DEFI_LOG | :QUALIFIER MATCH? (ALSO DEASSIGN/ALL |
| | 03 | 12 | 0446 | 654 | BNEQ | 70\$ | :NO, CHECK NEXT |
| | 0096 | 31 | 0448 | 655 | BRW | 170\$ | :YES, THEN BRANCH |
| 00000000'8F | 51 | D1 | 044B | 656 | CMPL | R1,#CLISK_DEFI_NAME | :QUALIFIER MATCH? |
| | 03 | 12 | 0452 | 657 | BNEQ | 80\$ | :NO, CHECK NEXT |
| | 009B | 31 | 0454 | 658 | BRW | 180\$ | :YES, THEN BRANCH |
| 00000000'8F | 51 | D1 | 0457 | 659 | CMPL | R1,#CLISK_DEFI_TRAN | :QUALIFIER MATCH? |
| | 81 | 12 | 045E | 660 | BNEQ | 10\$ | :NO, IGNORE IT |
| | 00B3 | 31 | 0460 | 661 | BRW | 190\$ | :YES, THEN BRANCH |
| | | | 0463 | 662 | | | |
| OC AB | 03 | D0 | 0463 | 663 | MOVL | #PSL\$C_USER,ACMODE(R8) | :SET USER MODE |
| | 0082 | 31 | 0467 | 664 | BRW | 171\$ | :GET NEXT TOKEN |
| OC AB | 02 | D0 | 046A | 665 | MOVL | #PSL\$C_SUPER,ACMODE(R8) | :SET SUPER MODE |
| | 7C | 11 | 046E | 666 | BRB | 171\$ | :GET NEXT TOKEN |
| OC AB | 01 | D0 | 0470 | 667 | MOVL | #PSL\$C_EXEC,ACMODE(R8) | :SET EXEC MODE |
| | 76 | 11 | 0474 | 668 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 0476 | 669 | | | |
| 55 | FB91 | CF | 9E | 0476 | MOVAB | LNMSPROCESS,R5 | :USE PROCESS LOGICAL NAME TABLE |
| | 54 | 85 | 9A | 047B | MOVZBL | (R5)+,R4 | : |
| 10 | AB | 54 | 7D | 047E | MOVQ | R4,TABNAM(R8) | :SAVE THE DESCRIPTOR |
| | 68 | 11 | 0482 | 673 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 0484 | 674 | | | |
| 55 | FB83 | CF | 9E | 0484 | MOVAB | LNMSPROCESS,R5 | :ASSUME /NOJOB |
| | 05 | 53 | 00 | E0 | BBS | #PTR V NEGATE-PTR_V_FLAGS,R3,137\$ | :BR IF /NOJOB |
| 55 | FB86 | CF | 9E | 048D | MOVAB | LNMSJOB,R5 | :USE JOB LOGICAL NAME TABLE |
| | 54 | 85 | 9A | 0492 | MOVZBL | (R5)+,R4 | : |
| 10 | AB | 54 | 7D | 0495 | MOVQ | R4,TABNAM(R8) | :SAVE THE DESCRIPTOR |
| | 51 | 11 | 0499 | 680 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 049B | 681 | | | |
| 55 | FB6C | CF | 9E | 049B | MOVAB | LNMSPROCESS,R5 | :ASSUME /NOGROUP |
| | 05 | 53 | 00 | E0 | BBS | #PTR V NEGATE-PTR_V_FLAGS,R3,147\$ | :BR IF /NOGROUP |
| 55 | FB77 | CF | 9E | 04A0 | MOVAB | LNMSGROUP,R5 | :USE GROUP LOGICAL NAME TABLE |
| | 54 | 85 | 9A | 04A9 | MOVZBL | (R5)+,R4 | : |
| 10 | AB | 54 | 7D | 04AC | MOVQ | R4,TABNAM(R8) | :SAVE THE DESCRIPTOR |
| | 3A | 11 | 04B0 | 687 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 04B2 | 688 | | | |
| 55 | FB55 | CF | 9E | 04B2 | MOVAB | LNMSPROCESS,R5 | :ASSUME /NOSYSTEM |
| | 05 | 53 | 00 | E0 | BBS | #PTR V NEGATE-PTR_V_FLAGS,R3,157\$ | :BR IF /NOSYSTEM |
| 55 | FB6A | CF | 9E | 04BB | MOVAB | LNMS\$SYSTEM,R5 | :USE SYSTEM LOGICAL NAME TABLE |
| | 54 | 85 | 9A | 04C0 | MOVZBL | (R5)+,R4 | : |
| 10 | AB | 54 | 7D | 04C3 | MOVQ | R4,TABNAM(R8) | :SAVE THE DESCRIPTOR |
| | 23 | 11 | 04C7 | 694 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 04C9 | 695 | | | |
| 55 | FB3E | CF | 9E | 04C9 | MOVAB | LNMSPROCESS,R5 | :ASSUME PROCESS LOGICAL NAME TABLE |
| | 54 | 85 | 9A | 04CE | MOVZBL | (R5)+,R4 | : |
| 10 | AB | 54 | 7D | 04D1 | MOVQ | R4,TABNAM(R8) | :SAVE THE DESCRIPTOR |
| | 14 | 53 | E8 | 04D5 | BLBS | R3,171\$ | :BRANCH IF NEGATED |
| | FB25 | 30 | 04D8 | 700 | BSBW | DCL\$GETDVAL | :GET THE TABLE NAME |
| 10 | AB | 51 | 7D | 04DB | MOVQ | R1,TABNAM(R8) | :SAVE IT AWAY |
| | 0B | 11 | 04DF | 702 | BRB | 171\$ | :GET NEXT TOKEN |
| | | | 04E1 | 703 | | | |
| 08 | AB | 05 | C8 | 04E1 | BISL | #LOG_M:DEF_M,QUAL(R8) | :ASSUME /LOG OR /ALL |

| | | | | | | | |
|-------------|-------|----|------|------------|------|--------------------------------|------------------------------------|
| 08 AB | 04 53 | E9 | 04E5 | 705 | BLBC | R3,171\$ | :BRANCH IF SO |
| | 01 | CA | 04E8 | 706 | BICL | #LOG_M,QUAL(R8) | :SET /NOLOG OR /NOALL |
| | FEF2 | 31 | 04EC | 707 171\$: | BRW | 10\$ | :GET NEXT TOKEN |
| | FEF2 | 31 | 04EF | 708 172\$: | BRW | 20\$ | :PROCESS THE TOKEN |
| | | | 04F2 | 709 | | | |
| 08 AB | 02 | C8 | 04F2 | 710 180\$: | BISL | #ATTR_M,QUAL(R8) | :MARK /NAME ATTRIBUTES SEEN |
| | 04 AB | D4 | 04F6 | 711 | CLRL | NAME_ATTR(R8) | :ZERO INITIAL ATTRIBUTES |
| | FO 53 | E8 | 04F9 | 712 | BLBS | R3,171\$ | :BRANCH IF NEGATED |
| | FB01' | 30 | 04FC | 713 182\$: | BSBW | DCL\$GETDVAL | :GET ITS VALUE |
| | 02 | 55 | 04FF | 714 | CMPB | R5,#PTR_K,QUALVALU | :SKIP IF NOT A QUALIFIER VALUE |
| | EB | 12 | 0502 | 715 | BNEQ | 172\$ | : |
| 43 8F | 62 | 91 | 0504 | 716 | CMPB | (R2),#A/C/ | :CONFINE KEYWORD? |
| | 06 | 12 | 0508 | 717 | BNEQ | 184\$ | :NO, THEN BRANCH |
| 04 AB | 02 | C8 | 050A | 718 | BISL | #LNMSM_CONFINE,NAME_ATTR(R8) | :SET THE ATTRIBUTE |
| | EC | 11 | 050E | 719 | BRB | 182\$ | :GET NEXT VALUE |
| 04 AB | 01 | C8 | 0510 | 720 184\$: | BISL | #LNMSM_NO_ALIAS,NAME_ATTR(R8) | :SET THE ATTRIBUTE |
| | E6 | 11 | 0514 | 721 | BRB | 182\$ | :GET NEXT VALUE |
| | | | 0516 | 722 | | | |
| 08 AB | 02 | C8 | 0516 | 723 190\$: | BISL | #ATTR_M,QUAL(R8) | :MARK /TRANSLATION ATTRIBUTES SEEN |
| | 68 | D4 | 051A | 724 | CLRL | TRAN_ATTR(R8) | :ZERO INITIAL ATTRIBUTES |
| | CD 53 | E8 | 051C | 725 | BLBS | R3,171\$ | :BRANCH IF NEGATED |
| | FADE' | 30 | 051F | 726 192\$: | BSBW | DCL\$GETDVAL | :GET ITS VALUE |
| | 02 | 55 | 0522 | 727 | CMPB | R5,#PTR_K,QUALVALU | :SKIP IF NOT A QUALIFIER VALUE |
| | C8 | 12 | 0525 | 728 | BNEQ | 172\$ | : |
| 43 8F | 62 | 91 | 0527 | 729 | CMPB | (R2),#A/C/ | :CONCEALED KEYWORD? |
| | 09 | 12 | 052B | 730 | BNEQ | 194\$ | :NO, THEN BRANCH |
| 68 00000100 | 8F | C8 | 052D | 731 | BISL | #LNMSM_CONCEALED,TRAN_ATTR(R8) | :SET THE ATTRIBUTE |
| | E9 | 11 | 0534 | 732 | BRB | 192\$ | :GET NEXT VALUE |
| 68 00000200 | 8F | C8 | 0536 | 733 194\$: | BISL | #LNMSM_TERMINAL,TRAN_ATTR(R8) | :SET THE ATTRIBUTE |
| | E0 | 11 | 053D | 734 | BRB | 192\$ | :GET NEXT VALUE |
| | | | 053F | 735 | | | |
| 18 AB | 51 | 7D | 053F | 736 200\$: | MOVQ | R1,LOGNAM(R8) | :GET FIRST PARAMETER DESCRIPTOR |
| | | 05 | 0543 | 737 210\$: | RSB | | : |
| | | | 0544 | 738 | | | |

```
0544 740 .SBTTL GET TRANSLATION ATTRIBUTES
0544 741 :
0544 742 : SUBROUTINE TO PROCESS TRANSLATION ATTRIBUTES
0544 743 :
0544 744 : R0 = QUALIFIER SEEN FLAG
0544 745 : R1/R2 = DESCR OF NEXT PARAMETER
0544 746 : R3 = TRANSLATION ATTRIBUTES
0544 747 : R5 = TYPE OF LAST TOKEN SEEN
0544 748 :
0544 749 GET_TRAN ATTR:
0544 750 CLRQ -(SP)
0546 751 10$: BSBW DCL$GETDVAL
0549 752 20$: CMPB #PTR_K_PARAMETR,R5
054C 753 : BLEQ 90$
054E 754 : MOVL #1,4(SP)
0552 755 : CLRL (SP)
0554 756 : BISL #ATTR_M,QUAL(R8)
0558 757 : BLBS R3,10$
055B 758 30$: BSBW DCL$GETDVAL
055E 759 : CMPB R5,#PTR_K_QUALVALU
0561 760 : BNEQ 20$
0563 761 : CMPB (R2),#^A/C/
0567 762 : BNEQ 40$
0569 763 : BISL #LNMSM_CONCEALED,(SP)
0570 764 : BRB 30$
0572 765 40$: BISL #LNMSM_TERMINAL,(SP)
0579 766 : BRB 30$
057B 767 :
057B 768 90$: POPL R3
057E 769 : POPL R0
0581 770 : RSB

7E 7C 0544 750 CLRQ -(SP)
FAB7' 30 0546 751 10$: BSBW DCL$GETDVAL
55 03 91 0549 752 20$: CMPB #PTR_K_PARAMETR,R5
2D 15 054C 753 : BLEQ 90$
04 AE 01 D0 054E 754 : MOVL #1,4(SP)
6E D4 0552 755 : CLRL (SP)
08 AB 02 C8 0554 756 : BISL #ATTR_M,QUAL(R8)
EB 53 E8 0558 757 : BLBS R3,10$
FAA2' 30 055B 758 30$: BSBW DCL$GETDVAL
02 55 91 055E 759 : CMPB R5,#PTR_K_QUALVALU
E6 12 0561 760 : BNEQ 20$
43 8F 62 91 0563 761 : CMPB (R2),#^A/C/
09 12 0567 762 : BNEQ 40$
6E 00000100 8F C8 0569 763 : BISL #LNMSM_CONCEALED,(SP)
E9 11 0570 764 : BRB 30$
6E 00000200 8F C8 0572 765 40$: BISL #LNMSM_TERMINAL,(SP)
E0 11 0579 766 : BRB 30$
53 8ED0 057B 767 :
50 8ED0 057B 768 90$: POPL R3
05 057E 769 : POPL R0
0581 770 : RSB

:GET TRAN ATTRIBUTES
:ASSUME QUALIFIER NOT SEEN
:GET NEXT DESCRIPTOR VALUE
:ITEM TYPE PARAMETER?
:IF LEQ END OF LINE OR PARAMETER
:MARK QUALIFIER SEEN
:RESET ATTRIBUTES
:MARK /TRANSLATION_ATTRIBUTES SEEN
:BRANCH IF NEGATED
:GET ITS VALUE
:SKIP IF NOT A QUALIFIER VALUE
:
:CONCEALED KEYWORD?
:NO, THEN BRANCH
:SET THE ATTRIBUTE
:GET NEXT VALUE
:SET THE ATTRIBUTE
:GET NEXT VALUE
:
:RETURN ATTRIBUTES AND STATUS
:
```

```
0582 772 .SBTTL CREATE LOGICAL NAME TABLE
0582 773
0582 774 :+ DCL$CRETABLE - CREATE LOGICAL NAME TABLE
0582 775
0582 776 THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE
0582 777 CREATE/NAME_TABLE COMMAND.
0582 778
0582 779 INPUTS:
0582 780
0582 781 R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
0582 782 R9 = ADDRESS OF SCRATCH STACK.
0582 783 R10 = BASE ADDRESS OF COMMAND WORK AREA.
0582 784 R11 = BASE ADDRESS OF PROCESS WORK AREA.
0582 785
0582 786 OUTPUTS:
0582 787
0582 788 THE SPECIFIED LOGICAL NAME TABLE IS CREATED.
0582 789
0582 790
0582 791 DCL$CRETABLE::
0582 792 CLRL -(R9)
0584 793 MOVZWL #DEF_PROT,-(R9)
0589 794
0589 795
0589 796 MOVL #PSL$C_SUPER,-(R9)
058C 797 CLRL -(R9)
058E 798 MOVL #LN$M_CREATE_IF,-(R9)
0595 799 MOVAB LN$PROCESS_DIRECTORY,R1
059A 800 MOVZBL (R1)+,R0
059D 801 MOVQ R0,-(R9)
05A0 802
05A0 803 10$: BSBW DCL$GETDVAL
05A3 804 20$: CMPB #PTR_K_PARAMETR,R5
05A6 805 BNEQ 30$
05A8 806 BRW 200$
05AB 807
05AB 808 30$: BSBW DCL$GETNVAL
05AE 809 CMPL R1,#CLISK_CRET_USER
05B5 810 BEQL 100$
05B7 811 CMPL R1,#CLISK_CRET_SUPE
05BE 812 BEQL 110$
05C0 813 CMPL R1,#CLISK_CRET_EXEC
05C7 814 BEQL 120$
05C9 815 CMPL R1,#CLISK_CRET_QUOT
05D0 816 BEQL 130$
05D2 817 CMPL R1,#CLISK_CRET_ATTR
05D9 818 BEQL 140$
05DB 819 CMPL R1,#CLISK_CRET_PARE
05E2 820 BEQL 150$
05E4 821 CMPL R1,#CLISK_CRET_PROT
05EB 822 BNEQ 40$
05ED 823 BRW 160$
05F0 824 40$: CMPL R1,#CLISK_CRET_LOG
05F7 825 BNEQ 10$
05F9 826 BRW 175$
05FC 827
05FC 828 100$: MOVL #PSL$C_USER,16(R9)

79 FF00 8F 3C 0582 792 CLRL -(R9)
79 01000000 8F 3C 0584 793 MOVZWL #DEF_PROT,-(R9)
79 51 FAAB CF 9E 0589 794
79 50 81 9A 0589 795
79 50 7D 0589 796 MOVL #PSL$C_SUPER,-(R9)
79 51 03 91 058C 797 CLRL -(R9)
79 03 12 058E 798 MOVL #LN$M_CREATE_IF,-(R9)
79 0124 31 0595 799 MOVAB LN$PROCESS_DIRECTORY,R1
79 51 03 9A 059A 800 MOVZBL (R1)+,R0
79 51 03 9A 059A 801 MOVQ R0,-(R9)
79 51 03 9A 059A 802
79 51 03 9A 059A 803 10$: BSBW DCL$GETDVAL
79 51 03 9A 059A 804 20$: CMPB #PTR_K_PARAMETR,R5
79 51 03 9A 059A 805 BNEQ 30$
79 51 03 9A 059A 806 BRW 200$
79 51 03 9A 059A 807
79 51 03 9A 059A 808 30$: BSBW DCL$GETNVAL
79 51 03 9A 059A 809 CMPL R1,#CLISK_CRET_USER
79 51 03 9A 059A 810 BEQL 100$
79 51 03 9A 059A 811 CMPL R1,#CLISK_CRET_SUPE
79 51 03 9A 059A 812 BEQL 110$
79 51 03 9A 059A 813 CMPL R1,#CLISK_CRET_EXEC
79 51 03 9A 059A 814 BEQL 120$
79 51 03 9A 059A 815 CMPL R1,#CLISK_CRET_QUOT
79 51 03 9A 059A 816 BEQL 130$
79 51 03 9A 059A 817 CMPL R1,#CLISK_CRET_ATTR
79 51 03 9A 059A 818 BEQL 140$
79 51 03 9A 059A 819 CMPL R1,#CLISK_CRET_PARE
79 51 03 9A 059A 820 BEQL 150$
79 51 03 9A 059A 821 CMPL R1,#CLISK_CRET_PROT
79 51 03 9A 059A 822 BNEQ 40$
79 51 03 9A 059A 823 BRW 160$
79 51 03 9A 059A 824 40$: CMPL R1,#CLISK_CRET_LOG
79 51 03 9A 059A 825 BNEQ 10$
79 51 03 9A 059A 826 BRW 175$
79 51 03 9A 059A 827
79 51 03 9A 059A 828 100$: MOVL #PSL$C_USER,16(R9)
```

| | | | | | | | | | |
|---------|----------|------|------|--------|--------|-----------------------|--------------------------------|----------------------------------|------------------------------|
| 10 A9 | 9E | 11 | 0600 | 829 | BRB | 10\$ | :GET NEXT TOKEN | | |
| | 02 | D0 | 0602 | 830 | MOVL | #PSL\$C_SUPER,16(R9) | :SET SUPER MODE | | |
| 10 A9 | 98 | 11 | 0606 | 831 | BRB | 10\$ | :GET NEXT TOKEN | | |
| | 01 | D0 | 0608 | 832 | MOVL | #PSL\$C_EXEC,16(R9) | :SET EXEC MODE | | |
| | 92 | 11 | 060C | 833 | BRB | 10\$ | :GET NEXT TOKEN | | |
| | | | 060E | 834 | | | | | |
| 0C A9 | D4 | 060E | 835 | 130\$: | CLRL | 12(R9) | :ASSUME /NOQUOTA | | |
| 10 53 | E8 | 0611 | 836 | | BLBS | R3,131\$ | :BRANCH IF SO | | |
| F9E9' | 30 | 0614 | 837 | | BSBW | DCL\$GETDVAL | :GET QUOTA VALUE | | |
| 52 | 51 | 7D | 0617 | 838 | MOVQ | R1,R2 | :COPY DESCRIPTOR | | |
| 51 | 01 | D0 | 061A | 839 | MOVL | #1,R1 | :SET DECIMAL RADIX | | |
| F9E0' | 30 | 061D | 840 | | BSBW | DCL\$CNVNOEDIT | :CONVERT NUMBER TO BINARY | | |
| 0C A9 | 51 | D0 | 0620 | 841 | MOVL | R1,12(R9) | :SAVE THE VALUE AWAY | | |
| FF79 | 31 | 0624 | 842 | 131\$: | BRW | 10\$ | :GET NEXT TOKEN | | |
| FF79 | 31 | 0627 | 843 | 132\$: | BRW | 20\$ | :PROCESS NEXT TOKEN | | |
| | | | 062A | 844 | | | | | |
| 08 A9 | 01000000 | 8F | D0 | 062A | 845 | 140\$: | MOVL | #LNMSM_CREATE_IF,8(R9) | :ASSUME /NOATTRIBUTES |
| | EF | 53 | E8 | 0632 | 846 | | BLBS | R3,131\$ | :BRANCH IF SO |
| | F9C8' | 30 | 0635 | 847 | 142\$: | BSBW | DCL\$GETDVAL | :GET ATTRIBUTE KEYWORD | |
| 02 | 55 | 91 | 0638 | 848 | CMPB | R5,#PTR_K_QUALVALU | :SKIP IF NOT A QUALIFIER VALUE | | |
| | EA | 12 | 063B | 849 | BNEQ | 132\$ | | | |
| 4E 8F | 62 | 91 | 063D | 850 | CMPB | (R2),#A/N/ | :NO_ALIAS KEYWORD? | | |
| | 06 | 12 | 0641 | 851 | BNEQ | 144\$ | :NO THEN BRANCH | | |
| 08 A9 | 01 | C8 | 0643 | 852 | BISL | #LNMSM_NO_ALIAS,8(R9) | :SET THE ATTRIBUTE | | |
| | EC | 11 | 0647 | 853 | BRB | 142\$ | :GET NEXT VALUE | | |
| 43 8F | 62 | 91 | 0649 | 854 | 144\$: | CMPB | (R2),#A/C/ | :CONFINE KEYWORD? | |
| | 06 | 12 | 064D | 855 | BNEQ | 144\$ | :NO THEN BRANCH | | |
| 08 A9 | 02 | C8 | 064F | 856 | BISL | #LNMSM_CONFINE,8(R9) | :SET THE ATTRIBUTE | | |
| | E0 | 11 | 0653 | 857 | BRB | 144\$ | :GET NEXT VALUE | | |
| 08 A9 | 01000000 | 8F | CA | 0655 | 858 | 146\$: | BICL | #LNMSM_CREATE_IF,8(R9) | :CLEAR THE ATTRIBUTE |
| | D6 | 11 | 065D | 859 | BRB | 142\$ | :GET NEXT VALUE | | |
| | | | 065F | 860 | | | | | |
| 51 | F9E1 | CF | 9E | 065F | 861 | 150\$: | MOVAB | LNMSPROCESS_DIRECTORY,R1 | :ASSUME /NOPARENT |
| | 50 | 81 | 9A | 0664 | 862 | | MOVZBL | (R1)+,R0 | |
| | 69 | 50 | 7D | 0667 | 863 | | MOVQ | R0,(R9) | :SAVE THE DESCRIPTOR |
| | B7 | 53 | E8 | 066A | 864 | | BLBS | R3,131\$ | :BRANCH IF SO |
| | F990' | 30 | 066D | 865 | | BSBW | DCL\$GETDVAL | :GET TABLE NAME | |
| | 69 | 51 | 7D | 0670 | 866 | | MOVQ | R1,(R9) | :SAVE THE DESCRIPTOR |
| | AF | 11 | 0673 | 867 | BRB | 131\$ | | :GET NEXT TOKEN | |
| | | | 0675 | 868 | | | | | |
| | | | 0675 | 869 | | | | | |
| | | | 0675 | 870 | | | | | |
| | F988' | 30 | 0675 | 871 | 160\$: | BSBW | DCL\$GETDVAL | :GET NEXT DESCRIPTOR VALUES | |
| 55 | 02 | 91 | 0678 | 872 | CMPB | #PTR_K_QUALVALU,R5 | :QUALIFIER VALUE? | | |
| | AA | 12 | 067B | 873 | BNEQ | 132\$ | :NO, ALL DONE WITH PROTECTION. | | |
| | | | 067D | 874 | | | | | |
| FA07 CF | 04 | 62 | 3A | 067D | 875 | | LOCC | (R2),#4,CLASS | :LOCATE PROTECTION CLASS |
| | | 3A | 13 | 0683 | 876 | | BEQL | 180\$ | :IF EQL INVALID CLASS |
| | | | 0685 | 877 | | | | | |
| | 50 | D7 | 0685 | 878 | DECL | R0 | | :CALCULATE STARTING BIT NUMBER | |
| 14 A9 | 58 | 04 | C5 | 0687 | 879 | | MULL3 | #4,R0,R8 | |
| | 58 | 0F | F0 | 068B | 880 | | INSV | #XF,R8,#4,20(R9) | :ASSUME NO ACCESS |
| | 54 | 02 | 91 | 0691 | 881 | | CMPB | #PTR_K_COLON,R4 | :PROTECTION VALUE SPECIFIED? |
| | | DF | 12 | 0694 | 882 | | BNEQ | 160\$ | :NO, TRY TO GET NEXT CLASS |
| | | | 0696 | 883 | | | | | |
| | F967' | 30 | 0696 | 884 | | BSBW | DCL\$GETDVAL | :GET PROTECTION VALUE DESCRIPTOR | |
| 57 | 51 | D0 | 0699 | 885 | MOVL | R1,R7 | | :SAVE LENGTH OF VALUE STRING | |


```
F9E4 CF 04 82 3A 069C 886
                23 13 069C 887 165$: LOCC (R2)+, #4, ACCESS ;LOCATE PROTECTION CODE
                50 58 D7 06A2 888 ;IF EQU INVALID PROTECTION CODE
                50 58 C0 06A4 889 ;CALCULATE RELATIVE BIT NUMBER
00 14 A9 50 58 E5 06A6 890 ;CALCULATE ACTUAL BIT NUMBER
                EB 57 F5 06A9 891 ;ALLOW SPECIFIED ACCESS
                C2 11 06AE 892 170$: SOBGTR R7, 165$ ;ANY MORE TO SCAN?
                06B1 893 ;NO, TRY TO GET NEXT CLASS
                06B3 894 ;
                06B3 895 ;
                06B3 896 ;
                18 A9 94 06B3 897 175$: CLRB 24(R9) ;ASSUME /LOG
                03 53 E9 06B6 898 ;IT IS /LOG. FLAG OK AS IS
                18 A9 96 06B9 899 ;IT IS /NOLOG. SET FLAG
                FF65 31 06BC 900 176$: BRW 131$ ;GET NEXT TOKEN
                06BF 901 ;
                06BF 902 180$: STATUS IVKEYW ;SET INVALID KEYWORD
                05 06C6 903 ;EXIT
                06C7 904 185$: STATUS IVPROT ;SET INVALID PROTECTION CODE
                05 06CE 905 ;EXIT
                06CF 906 ;
                79 51 7D 06CF 907 200$: MOVQ R1, -(R9) ;SAVE THE LOGICAL NAME DESR
                06D2 908 ;
                06D2 909 $CRELNT_S ATTR=16(R9),- ;CREATE THE TABLE
                06D2 910 -QUOTA=20(R9),-
                06D2 911 -TABNAM=(R9),-
                06D2 912 -PARTAB=8(R9),-
                06D2 913 -ACMODE=24(R9),-
                06D2 914 -PROMSK=28(R9)
                06EE 915 ;
                06EE 916 ;
                06EE 917 ;
                06EE 918 ;
                40 20 A9 E8 06EE 919 ;
                50 01 B1 06F2 920 ;
                09 12 06F5 921 ;
50 0003DE0B 8F D0 06F7 922 ;
                28 11 06FE 923 ;
                0700 924 ;
                50 0631 8F B1 0700 925 210$: CMPW #SS$_SUPERSEDE, R0 ;EXISTING TABLE SUPERSEDED?
                09 12 0705 926 ;NO, CHECK OTHER STATUS
50 0003DE13 8F D0 0707 927 ;YES, TELL USER
                18 11 070E 928 ;
                0710 929 ;
                50 06B1 8F B1 0710 930 220$: CMPW #SS$_LNMCREATED, R0 ;NEW TABLE CREATED?
                18 12 0715 931 ;NO, CHECK FOR CREATION ERROR
50 0003DE1B 8F D0 0717 932 ;ASSUME /SUPERSEDE SPECIFIED
10 A9 01000000 8F D3 071E 933 ;WAS /SUPERSEDE SPECIFIED?
                0D 12 0726 934 ;IF NOT, SKIP MESSAGE
                0728 935 ;
                69 9F 0728 936 270$: PUSHAB (R9) ;GET TABLE NAME DESCRIPTOR
                51 01 D0 072A 937 ;SET FAO COUNT
                F8D0 30 072D 938 ;OUTPUT MESSAGE
                03 11 0730 939 ;EXIT WITH STATUS NORMAL
                0732 940 ;
                07 50 E9 0732 941 280$: BLBC R0, 290$ ;BRANCH IF ERROR
                0735 942 285$: STATUS NORMAL ;RETURN SUCCESS
```

LOGICAL
V04-000

- LOGICAL NAME COMMANDS
CREATE LOGICAL NAME TABLE

L 16

16-SEP-1984 00:08:00 VAX/VMS Macro V04-00
4-SEP-1984 23:41:57 [DCL.SRC]LOGICAL.MAR;1

Page 23
(12)

05 0730 943 2908: RSB

:

```
073D 945 .SBTTL SHOW LOGICAL NAME EQUIVALENCES
073D 946 :+
073D 947 DCL$SHOWTRAN - SHOW LOGICAL NAME TRANSLATION
073D 948 :
073D 949 THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE SHOW LOGICAL
073D 950 NAME EQUIVALENCES DCL$ COMMAND.
073D 951 :
073D 952 INPUTS:
073D 953 :
073D 954 R8 = ADDRESS OF SCRATCH BUFFER DESCRIPTOR.
073D 955 R9 = ADDRESS OF SCRATCH STACK.
073D 956 R10 = BASE ADDRESS OF COMMAND WORK AREA.
073D 957 R11 = BASE ADDRESS OF PROCESS WORK AREA.
073D 958 :
073D 959 OUTPUTS:
073D 960 :
073D 961 THE SPECIFIED LOGICAL NAME EQUIVALENCE FROM THE PROCESS
073D 962 LOGICAL NAME TABLE IS WRITTEN TO THE OUTPUT STREAM.
073D 963 :-
073D 964 DCL$SHOWTRAN:: ;SHOW THE TRANSLATION FOR A NAME
073D 965 :
073D 966 :++
073D 967 Stack layout:
073D 968 :
073D 969 :-----:
073D 970 Table name
073D 971 descriptor
073D 972 :-----:
073D 973 Logical name
073D 974 descriptor
073D 975 :-----:
073D 976 Equival name
073D 977 descriptor
073D 978 :-----:
073D 979 Item list ...
073D 980 :
073D 981 :--
073D 982 :
073D 983 :
073D 984 Parse the command string.
073D 985 :
51 F8F3 CF 9E 073D 986 MOVAB LNMSDCL LOGICAL,R1 ;SET DEFAULT LOGICAL NAME TABLE
50 81 9A 0742 987 MOVZBL (R1)+,R0 ;
7E 50 7D 0745 988 MOVQ R0,-(SP) ;
0748 989 :
F8B5' 30 0748 990 10$: BSBW DCL$GETDVAL ;GET FIRST TOKEN
55 03 91 074B 991 CMPB #PTR_K_PARAMETER,R5 ;IS IT A PARAMETER
06 12 074E 992 BNEQ 15$ ;NO, THEN PROCESS /TABLE
F4 53 01 E0 0750 993 ASSUME PTR_V_KEYWORD EQ 21 ;
16 11 0754 995 BBS #1,R3,10$ ;IGNORE OPTION KEYWORD
0756 996 BRB 20$ ;PROCESS THE LOGICAL NAME
51 F8DA CF 9E 0756 997 15$: MOVAB LNMSDCL LOGICAL,R1 ;ASSUME /NOTABLE
50 81 9A 075B 998 MOVZBL (R1)+,R0 ;
6E 50 7D 075E 999 MOVQ R0,(SP) ;
E4 53 E8 0761 1000 BLBS R3,10$ ;BRANCH IF SO
F899' 30 0764 1001 BSBW DCL$GETDVAL ;GET TABLE NAME
```

| Address | Hex | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416 | Op417 | Op418 | Op419 | Op420 | Op421 | Op422 | Op423 | Op424 | Op425 | Op426 | Op427 | Op428 | Op429 | Op430 | Op431 | Op432 | Op433 | Op434 | Op435 | Op436 | Op437 | Op438 | Op439 | Op440 | Op441 | Op442 | Op443 | Op444 | Op445 | Op446 | Op447 | Op448 | Op449 | Op450 | Op451 | Op452 | Op453 | Op454 | Op455 | Op456 | Op457 | Op458 | Op459 | Op460 | Op461 | Op462 | Op463 | Op464 | Op46 |
|---------|-----|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|
|---------|-----|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|

- LOGICAL NAME COMMANDS
SHOW LOGICAL NAME EQUIVALENCES

C 1

16-SEP-1984 00:08:00
4-SEP-1984 23:41:57

VAX/VMS Macro V04-00
[DCL.SRC]LOGICAL.MAR;1

Page 26
(13)

| | | | | | | | | |
|----|----|------|----|------|------|--------|-------------|--------------------------------|
| SE | 2C | A7 | 9E | 0806 | 1059 | MOVAB | 44(R7),SP | ;RESTORE THE STACK |
| | 51 | 68 | 7D | 080A | 1060 | MOVQ | (R8),R1 | ;GET OUTPUT MESSAGE PARAMETERS |
| | | F7F0 | 30 | 080D | 1061 | BSBW | DCL\$MSGOUT | ;OUTPUT MESSAGE |
| | | | | 0810 | 1062 | STATUS | NORMAL | ;RETURN SUCCESS |
| | | | 05 | 0817 | 1063 | RSB | | |
| | | | | 0818 | 1064 | | | |
| | | | | 0818 | 1065 | .END | | |

LOGICAL
Symbol table

- LOGICAL NAME COMMANDS

D 1

16-SEP-1984 00:08:00 VAX/VMS Macro V04-00
4-SEP-1984 23:41:57 [DCL.SRC]LOGICAL.MAR;1

Page 27
(13)

```

$ST1 = 00000000
$ST2 = 00000006
ACCESS = 00000086 R 02
ACMODE = 0000000C
ATTR_M = 00000002
ATTR_V = 00000001
CLASS = 0000008A R 02
CLISK_ALLO_GENE ***** X 02
CLISK_CRET_ATTR ***** X 02
CLISK_CRET_EXEC ***** X 02
CLISK_CRET_LOG ***** X 02
CLISK_CRET_PARE ***** X 02
CLISK_CRET_PROT ***** X 02
CLISK_CRET_QUOT ***** X 02
CLISK_CRET_SUPE ***** X 02
CLISK_CRET_USER ***** X 02
CLISK_DEFI_EXEC ***** X 02
CLISK_DEFI_GROU ***** X 02
CLISK_DEFI_JOB ***** X 02
CLISK_DEFI_LOG ***** X 02
CLISK_DEFI_NAME ***** X 02
CLISK_DEFI_PROC ***** X 02
CLISK_DEFI_SUPE ***** X 02
CLISK_DEFI_SYST ***** X 02
CLISK_DEFI_TABL ***** X 02
CLISK_DEFI_TRAN ***** X 02
CLISK_DEFI_USER ***** X 02
CLIS_ALLOC = 0003DDE3
CLIS_IVKEYW = 00038060
CLIS_IVPROT = 00038070
CLIS_NORMAL = 00030001
CLIS_SUPERSEDE = 0003DDEB
CLIS_TABEXIST = 0003DE0B
CLIS_TABNOTFND = 0003DE1B
CLIS_TABSUPER = 0003DE13
COMMON_CRELNH = 000002AD R 02
COMMON_QUAL = 000003C7 R 02
DCL$ALLOCATE = 0000008E RG 02
DCL$ASSIGN = 000001A3 RG 02
DCL$CNVNOEDIT ***** X 02
DCL$COMPRESS ***** X 02
DCL$COMPSTRING ***** X 02
DCL$CREATE_OUTPUT ***** X 02
DCL$CRETABLE = 00000582 RG 02
DCL$DEALLOCAT = 00000314 RG 02
DCL$DEASSIGN = 0000033B RG 02
DCL$DEFINE = 00000231 RG 02
DCL$FORMMSG ***** X 02
DCL$GETDVAL ***** X 02
DCL$GETNVAL ***** X 02
DCL$MSGOUT ***** X 02
DCL$OPEN_OUTPUT ***** X 02
DCL$RESTORE_OUTPUT ***** X 02
DCL$SHOWTRAN = 0000073D RG 02
DEF_M = 00000004
DEF_PROT = 0000FF00
DEF_V = 00000002

```

```

EQUAM = 00000020
GET_TRAN_ATTR = 00000544 R 02
LNMSDCL LOGICAL = 00000034 R 02
LNMSFILE DEV = 0000005A R 02
LNMSGROUP = 0000001F R 02
LNMSJOB = 00000017 R 02
LNMSM_CONCEALED = 00000100
LNMSM_CONFINE = 00000002
LNMSM_CREATE_IF = 01000000
LNMSM_NO_ALIAS = 00000001
LNMSM_TERMINAL = 00000200
LNMSPROCESS = 0000000B R 02
LNMSPROCESS_DIRECTORY = 00000044 R 02
LNMSYSTEM = 00000029 R 02
LNMS_ATTRIBUTES = 00000003
LNMS_STRING = 00000002
LNMS_TABLE = 00000004
LOGICALMSG = 00000071 R 02
LOGNAM = 00000018
LOG_M = 00000001
LOG_V = 00000000
NAME_ATTR = 00000004
OUTPUTNAM = 00000000 R 02
PRC_B_CONTINUE = 000000F3
PRC_B_DEFRADIX = 000000AE
PRC_B_EXMDEPMOD = 000000AD
PRC_B_EXMDEPWID = 000000AC
PRC_B_EXONLYL = 0000012D
PRC_B_FLAGS2 = 000000AF
PRC_B_IMGFLAG = 0C000078
PRC_B_OUTFLAGS = 0000012C
PRC_B_PROMPTLEN = 000000F0
PRC_C_LENGTH = 00000534
PRC_G_COMMANDS = 00000133
PRC_G_PROMPT = 000000F4
PRC_K_LENGTH = 00000534
PRC_L_CURRKEY = 00000048
PRC_L_EXMDEPADR = 000000A8
PRC_L_EXTARG = 00000094
PRC_L_EXTBLK = 0000008C
PRC_L_EXTCOD = 0000009C
PRC_L_EXTHND = 00000090
PRC_L_EXTPRM = 00000098
PRC_L_IDFLNK = 000000BC
PRC_L_IMGACTSTS = 00000080
PRC_L_INDCLOCK = 0000007C
PRC_L_INDEPTH = 0000005C
PRC_L_INDFAB = 0000001C
PRC_L_INDINPRAB = 00000014
PRC_L_INDOUTRAB = 00000018
PRC_L_INPRAB = 00000008
PRC_L_LASTKEY = 0000004C
PRC_L_LSTSTATUS = 000000B0
PRC_L_ONCTLY = 000000B8
PRC_L_ONERROR = 0000006C
PRC_L_OUTOFBAND = 000000B4
PRC_L_OUTRAB = 0000000C

```

LOGICAL
Symbol table

- LOGICAL NAME COMMANDS

E 1

16-SEP-1984 00:08:00 VAX/VMS Macro V04-00
4-SEP-1984 23:41:57 [DCL.SRC]LOGICAL.MAR;1

Page 28
(13)

```

PRC_L_OUTRABCTX      00000118
PRC_L_PPFLIST        00000070
PRC_L_RECALLPTR      0000012F
PRC_L_RESTART        00000058
PRC_L_SAVAP          00000000
PRC_L_SAVFP          00000004
PRC_L_SEVERITY       00000050
PRC_L_SPWN           000000C0
PRC_L_STACKLM        000000A4
PRC_L_STACKPT        000000A0
PRC_L_STATUS         00000054
PRC_L_STS            00000084
PRC_L_STV            00000088
PRC_L_SYMBOL         00000060
PRC_L_TMBX           00000074
PRC_L_TRMLIST        00000010
PRC_Q_ALLOCREG       00000020
PRC_Q_COMMAND        000000E0
PRC_Q_FLUSHTIME      000000D0
PRC_Q_GLOBAL         00000028
PRC_Q_IMAGENAME      000000D8
PRC_Q_KEYPAD         00000040
PRC_Q_LABEL          00000030
PRC_Q_LOCAL          00000038
PRC_Q_SAVEPRIV       000000E8
PRC_T_OUTDVI         0000011C
PRC_W_ASTIOSB        000000C6
PRC_W_ASTRETN        000000C8
PRC_W_ASTSTATUS      000000C4
PRC_W_ATTMBX         0000007A
PRC_W_FLAGS          00000068
PRC_W_INPCHAN        00000064
PRC_W_ONLEVEL        0000006A
PRC_W_OUTIFI         00000114
PRC_W_OUTISI         00000116
PRC_W_OUTMBXCHN      000000CA
PRC_W_OUTMBXREF      000000CE
PRC_W_OUTMBXSIZ      000000CC
PRC_W_PMPCTCTL       000000F1
PRC_W_WAITIOSB       00000066
PSL$C_EXEC           = 00000001
PSL$C_SUPER          = 00000002
PSL$C_USER           = 00000003
PTR_B_LEVEL          00000004
PTR_B_NUMBER         00000005
PTR_B_PARMCNT        00000006
PTR_B_VALUE          00000000
PTR_C_LENGTH         = 0000000C
PTR_K_COLON          = 00000002
PTR_K_COMMA          = 00000005
PTR_K_ENDLINE        = 00000004
PTR_K_LENGTH         = 0000000C
PTR_K_PARAMETER      = 00000003
PTR_K_QUALVALU       = 00000002
PTR_L_DESCR          00000000
PTR_L_ENTITY         00000008
PTR_V_FLAGS          = 00000014

```

```

PTR_V_KEYWORD        = 00000015
PTR_V_NEGATE         = 00000014
QUAL                 = 00000008
SS$_LNMCREATED       = 000006B1
SS$_NORMAL           = 00000001
SS$_NOSUCHDEV        = 00000908
SS$_SUPERSEDE        = 00000631
SY$ALLOC             ***** GX 02
SY$SCRELNM           ***** GX 02
SY$SCRELNT           ***** GX 02
SY$SDALLOC           ***** GX 02
SY$SDELLNM           ***** GX 02
SY$SFAO              ***** X 02
SY$STRNLNM           ***** GX 02
TABNAM               = 00000010
TESTOUT              00000391 R 02
TRAN_ATTR            = 00000000
UNDEFINED            00000067 R 02
WRK_B_CMDOPT         FFFFFFFC3
WRK_B_MAXPARM        FFFFFFFD0
WRK_B_MINPARM        FFFFFFFD1
WRK_B_PARMCNT        FFFFFFFCE
WRK_B_PARMSUM        FFFFFFFCF
WRK_B_RECALLCNT      FFFFFFFC5
WRK_B_VALLEV         FFFFFFFC4
WRK_B_VERBTYP        FFFFFFFC2
WRK_C_INPBUFFSIZ     = 00000100
WRK_C_LENGTH         FFFFFFF486
WRK_G_BUFFER         FFFFFFF492
WRK_G_INPBUF         FFFFFFF896
WRK_G_RESULT         FFFFFFF9B6
WRK_K_LENGTH         FFFFFFF486
WRK_L_CHARPTR        FFFFFFF48E
WRK_L_DISALLOW       FFFFFFFE6
WRK_L_ERRORRTN       FFFFFFF9AE
WRK_L_EXPANDPTR      FFFFFFF486
WRK_L_IMAGE          FFFFFFFE2
WRK_L_MARKPTR        FFFFFFF48A
WRK_L_PAROUT         FFFFFFFD2
WRK_L_PMPADDR        FFFFFFF9A2
WRK_L_PROMPTRTN      FFFFFFF9A6
WRK_L_PROPTR         FFFFFFFC6
WRK_L_QUABLK         FFFFFFFCA
WRK_L_READRTN        FFFFFFF9AA
WRK_L_RECALLPTR      FFFFFFFEA
WRK_L_RSLND          FFFFFFFB6
WRK_L_RSLNXT         FFFFFFFBA
WRK_L_SAVAP          FFFFFFFF8
WRK_L_SAVFP          FFFFFFFFC
WRK_L_SAVSP          FFFFFFFF4
WRK_L_SIGNALRTN      FFFFFFFD6
WRK_L_SPECRTN        FFFFFFF9B2
WRK_L_TAB_VEC        FFFFFFFDE
WRK_L_VERB           FFFFFFFBE
WRK_W_FLAGS          FFFFFFFF0
WRK_W_FLAGS2         FFFFFFFF2
WRK_W_IMGCHAN        FFFFFFFEE

```


LOGICAL
Symbol table

- LOGICAL NAME COMMANDS

F 1

16-SEP-1984 00:08:00 VAX/VMS Macro V04-00
4-SEP-1984 23:41:57 [DCL.SRC]LOGICAL.MAR;1

Page 29
(13)

WRK_W_PMPLEN

FFFFFF99E

+-----+
! Psect synopsis !
+-----+

| PSECT name | Allocation | PSECT No. | Attributes |
|------------|-------------------|-----------|---|
| . ABS . | 00000000 (0.) | 00 (0.) | NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE |
| \$AB\$\$ | FFFFFFFC (0.) | 01 (1.) | NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE |
| DCL\$ZCODE | 00000818 (2072.) | 02 (2.) | NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE |

+-----+
! Performance indicators !
+-----+

| Phase | Page faults | CPU Time | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization | 15 | 00:00:00.08 | 00:00:01.26 |
| Command processing | 101 | 00:00:00.65 | 00:00:05.99 |
| Pass 1 | 331 | 00:00:13.05 | 00:00:33.89 |
| Symbol table sort | 0 | 00:00:01.69 | 00:00:04.36 |
| Pass 2 | 183 | 00:00:03.27 | 00:00:09.90 |
| Symbol table output | 27 | 00:00:00.18 | 00:00:00.64 |
| Psect synopsis output | 2 | 00:00:00.03 | 00:00:00.05 |
| Cross-reference output | 0 | 00:00:00.00 | 00:00:00.00 |
| Assembler run totals | 659 | 00:00:18.96 | 00:00:56.11 |

The working set limit was 1500 pages.
69087 bytes (135 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 980 non-local and 104 local symbols.
1065 source lines were read in Pass 1, producing 20 object records in Pass 2.
43 pages of virtual memory were used to define 28 macros.

+-----+
! Macro library statistics !
+-----+

| Macro library name | Macros defined |
|---------------------------------------|----------------|
| -\$255\$DUA28:[SYSLIB]SYSBLDMLB.MLB;1 | 0 |
| -\$255\$DUA28:[DCL.OBJ]DCL.MLB;1 | 6 |
| -\$255\$DUA28:[SYS.OBJ]LIB.MLB;1 | 0 |
| -\$255\$DUA28:[SYSLIB]STARLET.MLB;2 | 16 |
| TOTALS (all libraries) | 22 |

1132 GETS were required to define 22 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:LOGICAL/OBJ=OBJ\$:LOGICAL MSRC\$:LOGICAL/UPDATE=(ENH\$:LOGICAL)+EXECMLS/LIB+LIB\$:DCL/LIB+SYSS\$LIBRARY:SYSBLDMLB/LIB

0071 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

INQUIRE
LIS

LEXICON
LIS

KEYPAD
LIS

LOGICAL
LIS

0072 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 001 | 002 | 003 | 004 | 005 | 006 | 007 | 008 | 009 | 010 | 011 | 012 | 013 | 014 | 015 | 016 | 017 | 018 | 019 | 020 | 021 | 022 | 023 | 024 | 025 | 026 | 027 | 028 | 029 | 030 | 031 | 032 | 033 | 034 | 035 | 036 | 037 | 038 | 039 | 040 | 041 | 042 | 043 | 044 | 045 | 046 | 047 | 048 | 049 | 050 | 051 | 052 | 053 | 054 | 055 | 056 | 057 | 058 | 059 | 060 | 061 | 062 | 063 | 064 | 065 | 066 | 067 | 068 | 069 | 070 | 071 | 072 | 073 | 074 | 075 | 076 | 077 | 078 | 079 | 080 | 081 | 082 | 083 | 084 | 085 | 086 | 087 | 088 | 089 | 090 | 091 | 092 | 093 | 094 | 095 | 096 | 097 | 098 | 099 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 | 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 | 221 | 222 | 223 | 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 | 233 | 234 | 235 | 236 | 237 | 238 | 239 | 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 | 251 | 252 | 253 | 254 | 255 | 256 | 257 | 258 | 259 | 260 | 261 | 262 | 263 | 264 | 265 | 266 | 267 | 268 | 269 | 270 | 271 | 272 | 273 | 274 | 275 | 276 | 277 | 278 | 279 | 280 | 281 | 282 | 283 | 284 | 285 | 286 | 287 | 288 | 289 | 290 | 291 | 292 | 293 | 294 | 295 | 296 | 297 | 298 | 299 | 300 | 301 | 302 | 303 | 304 | 305 | 306 | 307 | 308 | 309 | 310 | 311 | 312 | 313 | 314 | 315 | 316 | 317 | 318 | 319 | 320 | 321 | 322 | 323 | 324 | 325 | 326 | 327 | 328 | 329 | 330 | 331 | 332 | 333 | 334 | 335 | 336 | 337 | 338 | 339 | 340 | 341 | 342 | 343 | 344 | 345 | 346 | 347 | 348 | 349 | 350 | 351 | 352 | 353 | 354 | 355 | 356 | 357 | 358 | 359 | 360 | 361 | 362 | 363 | 364 | 365 | 366 | 367 | 368 | 369 | 370 | 371 | 372 | 373 | 374 | 375 | 376 | 377 | 378 | 379 | 380 | 381 | 382 | 383 | 384 | 385 | 386 | 387 | 388 | 389 | 390 | 391 | 392 | 393 | 394 | 395 | 396 | 397 | 398 | 399 | 400 | 401 | 402 | 403 | 404 | 405 | 406 | 407 | 408 | 409 | 410 | 411 | 412 | 413 | 414 | 415 | 416 | 417 | 418 | 419 | 420 | 421 | 422 | 423 | 424 | 425 | 426 | 427 | 428 | 429 | 430 | 431 | 432 | 433 | 434 | 435 | 436 | 437 | 438 | 439 | 440 | 441 | 442 | 443 | 444 | 445 | 446 | 447 | 448 | 449 | 450 | 451 | 452 | 453 | 454 | 455 | 456 | 457 | 458 | 459 | 460 | 461 | 462 | 463 | 464 | 465 | 466 | 467 | 468 | 469 | 470 | 471 | 472 | 473 | 474 | 475 | 476 | 477 | 478 | 479 | 480 | 481 | 482 | 483 | 484 | 485 | 486 | 487 | 488 | 489 | 490 | 491 | 492 | 493 | 494 | 495 | 496 | 497 | 498 | 499 | 500 | 501 | 502 | 503 | 504 | 505 | 506 | 507 | 508 | 509 | 510 | 511 | 512 | 513 | 514 | 515 | 516 | 517 | 518 | 519 | 520 | 521 | 522 | 523 | 524 | 525 | 526 | 527 | 528 | 529 | 530 | 531 | 532 | 533 | 534 | 535 | 536 | 537 | 538 | 539 | 540 | 541 | 542 | 543 | 544 | 545 | 546 | 547 | 548 | 549 | 550 | 551 | 552 | 553 | 554 | 555 | 556 | 557 | 558 | 559 | 560 | 561 | 562 | 563 | 564 | 565 | 566 | 567 | 568 | 569 | 570 | 571 | 572 | 573 | 574 | 575 | 576 | 577 | 578 | 579 | 580 | 581 | 582 | 583 | 584 | 585 | 586 | 587 | 588 | 589 | 590 | 591 | 592 | 593 | 594 | 595 | 596 | 597 | 598 | 599 | 600 | 601 | 602 | 603 | 604 | 605 | 606 | 607 | 608 | 609 | 610 | 611 | 612 | 613 | 614 | 615 | 616 | 617 | 618 | 619 | 620 | 621 | 622 | 623 | 624 | 625 | 626 | 627 | 628 | 629 | 630 | 631 | 632 | 633 | 634 | 635 | 636 | 637 | 638 | 639 | 640 | 641 | 642 | 643 | 644 | 645 | 646 | 647 | 648 | 649 | 650 | 651 | 652 | 653 | 654 | 655 | 656 | 657 | 658 | 659 | 660 | 661 | 662 | 663 | 664 | 665 | 666 | 667 | 668 | 669 | 670 | 671 | 672 | 673 | 674 | 675 | 676 | 677 | 678 | 679 | 680 | 681 | 682 | 683 | 684 | 685 | 686 | 687 | 688 | 689 | 690 | 691 | 692 | 693 | 694 | 695 | 696 | 697 | 698 | 699 | 700 | 701 | 702 | 703 | 704 | 705 | 706 | 707 | 708 | 709 | 710 | 711 | 712 | 713 | 714 | 715 | 716 | 717 | 718 | 719 | 720 | 721 | 722 | 723 | 724 | 725 | 726 | 727 | 728 | 729 | 730 | 731 | 732 | 733 | 734 | 735 | 736 | 737 | 738 | 739 | 740 | 741 | 742 | 743 | 744 | 745 | 746 | 747 | 748 | 749 | 750 | 751 | 752 | 753 | 754 | 755 | 756 | 757 | 758 | 759 | 760 | 761 | 762 | 763 | 764 | 765 | 766 | 767 | 768 | 769 | 770 | 771 | 772 | 773 | 774 | 775 | 776 | 777 | 778 | 779 | 780 | 781 | 782 | 783 | 784 | 785 | 786 | 787 | 788 | 789 | 790 | 791 | 792 | 793 | 794 | 795 | 796 | 797 | 798 | 799 | 800 | 801 | 802 | 803 | 804 | 805 | 806 | 807 | 808 | 809 | 810 | 811 | 812 | 813 | 814 | 815 | 816 | 817 | 818 | 819 | 820 | 821 | 822 | 823 | 824 | 825 | 826 | 827 | 828 | 829 | 830 | 831 | 832 | 833 | 834 | 835 | 836 | 837 | 838 | 839 | 840 | 841 | 842 | 843 | 844 | 845 | 846 | 847 | 848 | 849 | 850 | 851 | 852 | 853 | 854 | 855 | 856 | 857 | 858 | 859 | 860 | 861 | 862 | 863 | 864 | 865 | 866 | 867 | 868 | 869 | 870 | 871 | 872 | 873 | 874 | 875 | 876 | 877 | 878 | 879 | 880 | 881 | 882 | 883 | 884 | 885 | 886 | 887 | 888 | 889 | 890 | 891 | 892 | 893 | 894 | 895 | 896 | 897 | 898 | 899 | 900 | 901 | 902 | 903 | 904 | 905 | 906 | 907 | 908 | 909 | 910 | 911 | 912 | 913 | 914 | 915 | 916 | 917 | 918 | 919 | 920 | 921 | 922 | 923 | 924 | 925 | 926 | 927 | 928 | 929 | 930 | 931 | 932 | 933 | 934 | 935 | 936 | 937 | 938 | 939 | 940 | 941 | 942 | 943 | 944 | 945 | 946 | 947 | 948 | 949 | 950 | 951 | 952 | 953 | 954 | 955 | 956 | 957 | 958 | 959 | 960 | 961 | 962 | 963 | 964 | 965 | 966 | 967 | 968 | 969 | 970 | 971 | 972 | 973 | 974 | 975 | 976 | 977 | 978 | 979 | 980 | 981 | 982 | 983 | 984 | 985 | 986 | 987 | 988 | 989 | 990 | 991 | 992 | 993 | 994 | 995 | 996 | 997 | 998 | 999 | 1000 | 1001 | 1002 | 1003 | 1004 | 1005 | 1006 | 1007 | 1008 | 1009 | 1010 | 1011 | 1012 | 1013 | 1014 | 1015 | 1016 | 1017 | 1018 | 1019 | 1020 | 1021 | 1022 | 1023 | 1024 | 1025 | 1026 | 1027 | 1028 | 1029 | 1030 | 1031 | 1032 | 1033 | 1034 | 1035 | 1036 | 1037 | 1038 | 1039 | 1040 | 1041 | 1042 | 1043 | 1044 | 1045 | 1046 | 1047 | 1048 | 1049 | 1050 | 1051 | 1052 | 1053 | 1054 | 1055 | 1056 | 1057 | 1058 | 1059 | 1060 | 1061 | 1062 | 1063 | 1064 | 1065 | 1066 | 1067 | 1068 | 1069 | 1070 | 1071 | 1072 | 1073 | 1074 | 1075 | 1076 | 1077 | 1078 | 1079 | 1080 | 1081 | 1082 | 1083 | 1084 | 1085 | 1086 | 1087 | 1088 | 1089 | 1090 | 1091 | 1092 | 1093 | 1094 | 1095 | 1096 | 1097 | 1098 | 1099 | 1100 | 1101 | 1102 | 1103 | 1104 | 1105 | 1106 | 1107 | 1108 | 1109 | 1110 | 1111 | 1112 | 1113 | 1114 | 1115 | 1116 | 1117 | 1118 | 1119 | 1120 | 1121 | 1122 | 1123 | 1124 | 1125 | 1126 | 1127 | 1128 | 1129 | 1130 | 1131 | 1132 | 1133 | 1134 | 1135 | 1136 | 1137 | 1138 | 1139 | 1140 | 1141 | 1142 | 1143 | 1144 | 1145 | 1146 | 1147 | 1148 | 1149 | 1150 | 1151 | 1152 | 1153 | 1154 | 1155 | 1156 | 1157 | 1158 | 1159 | 1160 | 1161 | 1162 | 1163 | 1164 | 1165 | 1166 | 1167 | 1168 | 1169 | 1170 | 1171 | 1172 | 1173 | 1174 | 1175 | 1176 | 1177 | 1178 | 1179 | 1180 | 1181 | 1182 | 1183 | 1184 | 1185 | 1186 | 1187 | 1188 | 1189 | 1190 | 1191 | 1192 | 1193 | 1194 | 1195 | 1196 | 1197 | 1198 | 1199 | 1200 | 1201 | 1202 | 1203 | 1204 | 1205 | 1206 | 1207 | 1208 | 1209 | 1210 | 1211 | 1212 | 1213 | 1214 | 1215 | 1216 | 1217 | 1218 | 1219 | 1220 | 1221 | 1222 | 1223 | 1224 | 1225 | 1226 | 1227 | 1228 | 1229 | 1230 | 1231 | 1232 | 1233 | 1234 | 1235 | 1236 | 1237 | 1238 | 1239 | 1240 | 1241 | 1242 | 1243 | 1244 | 1245 | 1246 | 1247 | 1248 | 1249 | 1250 | 1251 | 1252 | 1253 | 1254 | 1255 | 1256 | 1257 | 1258 | 1259 | 1260 | 1261 | 1262 | 1263 | 1264 | 1265 | 1266 | 1267 | 1268 | 1269 | 1270 | 1271 | 1272 | 1273 | 1274 | 1275 | 1276 | 1277 | 1278 | 1279 | 1280 | 1281 | 1282 | 1283 | 1284 | 1285 | 1286 | 1287 | 1288 | 1289 | 1290 | 1291 | 1292 | 1293 | 1294 | 1295 | 1296 | 1297 | 1298 | 1299 | 1300 | 1301 | 1302 | 1303 | 1304 | 1305 | 1306 | 1307 | 1308 | 1309 | 1310 | 1311 | 1312 | 1313 | 1314 | 1315 | 1316 | 1317 | 1318 | 1319 | 1320 | 1321 | 1322 | 1323 | 1324 | 1325 | 1326 | 1327 | 1328 | 1329 | 1330 | 1331 | 1332 | 1333 | 1334 | 1335 | 1336 | 1337 | 1338 | 1339 | 1340 | 1341 | 1342 | 1343 | 1344 | 1345 | 1346 | 1347 | 1348 | 1349 | 1350 | 1351 | 1352 | 1353 | 1354 | 1355 | 1356 | 1357 | 1358 | 1359 | 1360 | 1361 | 1362 | 1363 | 1364 | 1365 | 1366 | 1367 | 1368 | 1369 | 1370 | 1371 | 1372 | 1373 | 1374 | 1375 | 1376 | 1377 | 1378 | 1379 | 1380 | 1381 | 1382 | 1383 | 1384 | 1385 | 1386 | 1387 | 1388 | 1389 | 1390 | 1391 | 1392 | 1393 | 1394 | 1395 | 1396 | 1397 | 1398 | 1399 | 1400 | 1401 | 1402 | 1403 | 1404 | 1405 | 1406 | 1407 | 1408 | 1409 | 1410 | 1411 | 1412 | 1413 | 1414 | 1415 | 1416 | 1417 | 1418 | 1419 | 1420 | 1421 | 1422 | 1423 | 1424 | 1425 | 1426 | 1427 | 1428 | 1429 | 1430 | 1431 | 1432 | 1433 | 1434 | 1435 | 1436 | 1437 | 1438 | 1439 | 1440 | 1441 | 1442 | 1443 | 1444 | 1445 | 1446 | 1447 | 1448 | 1449 | 1450 | 1451 | 1452 | 1453 | 1454 | 1455 | 1456 | 1457 | 1458 | 1459 | 1460 | 1461 | 1462 | 1463 | 1464 | 1465 | 1466 | 1467 | 1468 | 1469 | 1470 | 1471 | 1472 | 1473 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|